Numerical Solution of Linear and Nonlinear Eigenvalue Problems

Akinola, Richard

Award date:
2010

Awarding institution:
University of Bath

Numerical Solution of Linear and Nonlinear Eigenvalue Problems

submitted by

Richard Ọlatokunbọ Akinọla

for the degree of Doctor of Philosophy

of the

University of Bath

Department of Mathematical Sciences

May 2010

SUMMARY

Given a real parameter-dependent matrix, we obtain an algorithm for computing the value of the parameter and corresponding eigenvalue for which two eigenvalues of the matrix coalesce to form a 2-dimensional Jordan block. Our algorithms are based on extended versions of the implicit determinant method of Spence and Poulton [55]. We consider when the eigenvalue is both real and complex, which results in solving systems of nonlinear equations by Newton's or the Gauss-Newton method. Our algorithms rely on good initial guesses, but if these are available, we obtain quadratic convergence.

Next, we describe two quadratically convergent algorithms for computing a nearby defective matrix which are cheaper than already known ones. The first approach extends the implicit determinant method in [55] to find parameter values for which a certain Hermitian matrix is singular subject to a constraint. This results in using Newton's method to solve a real system of three nonlinear equations. The second approach involves simply writing down all the nonlinear equations and solving a real over-determined system using the Gauss-Newton method. We only consider the case where the nearest defective matrix is real.

Finally, we consider the computation of an algebraically simple complex eigenpair of a nonsymmetric matrix where the eigenvector is normalised using the natural 2-norm, which produces only a single real normalising equation. We obtain an under-determined system of nonlinear equations which is solved by the Gauss-Newton method. We show how to obtain an equivalent square linear system of equations for the computation of the desired eigenpairs. This square system is exactly what would have been obtained if we had ignored the non uniqueness and non differentiability of the normalisation.

ACKNOWLEDGEMENTS

Thank you Lord for saving me. Indeed, ”it is not of him that willeth, neither of him that runneth, but of God that showeth mercies.” Blessed be your name for bringing me to the end of my academic career against all odds.

My profound appreciation goes to my supervisor, Professor Alastair Spence for his assistance, gainful discussions, patience and positive feedbacks during the period of my Ph.D. To Professor Ivan Graham, for his comments during the review stages, they really helped to improve this work and Dr. Françoise Tisseur (University of Manchester) for reading this thesis.

This Ph.D. would not have been possible if not for the University of Bath studentship I received to which I say ”thanks with much 'muchness' ” for this rare priviledge. Thanks to the Vice-Chancellor of the University of Jos, as well as other colleagues of the Department of Mathematics, University of Jos, Nigeria. Prof. L. S. O. Liverpool for his immense support.

Many thanks to my office mates: Lisema for being a brother, Dave, Aidan, Simon, Claire, Robert. To Melina, Zhivko, Chris, Mr Cooper & Gabrielle. Thanks to the staff and computer support team of the department for ever willing to help. Kudos to Bro & Sis. Okor and Pastor Juliana for interceding on my behalf. Prof. Mike Threadgill, thanks for sharing the burden during the difficult moments. To my siblings: Ọlaronke, the families of Ọlayinka and Pastor 'Niyi for their prayers when the ship almost capsized. Much thanks to my Dad, Mr S. O. Akinọla for teaching me how to read and enduring all the pains my studies cost him. Ẹ sun re, Mọmọ mi. To Oyekẹmi Bọlarinwa-BBK for been there and all those far and near whom for reason of space I can't mention your names, I say 'ẹ seun=thank you.' I must not forget the AIMS family South Africa, and all those whose destiny is co-optied with this Ph.D.

Emi lẹni t'aiye ti ro, wipe oun le dan nkan nkan se. Sugbọn mori anu Rẹ gba, Olu Ọrun lo ba mi se. Ẹẹru Ọlọrun ba mi 2×, oun to ba ti pinu lọkọn rẹ, ko sẹni to le daduro.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF NOTATIONS

We present a list of frequently used notations in this thesis. Real and complex scalars are written with Greek letters while vectors and matrices are boldfaced.

Symbols	**Definitions**
$\Lambda_\varepsilon(\mathbf{A})$	ε-pseudospectrum of $\mathbf{A}$ 10
$\|.\|_2$	2-norm 79
$\mathbf{A}^H$	Hermitian transpose of $\mathbf{A}$ 9
σ_j	jth singular value 9
$\mathbf{I}$	Identity matrix 5
$\mathbf{O}$	Zero matrix 121
$\mathbf{e}_k$	The kth column of an identity matrix 27
$\mathbb{N}$	The set of all natural numbers. 20
$\mathbb{R}$	The set of all real numbers. 9
$\mathbb{C}$	The set of all complex numbers. 5
$\mathbb{C}^n$	An n component complex vector. 5
$\mathbb{C}^{n\times n}$	The set of all n by n complex matrices with complex entries. 80
$\mathbb{R}^n$	An n component real vector. 11
$\mathbb{R}^{n\times n}$	The set of all n by n real matrices with real entries. 5
$\kappa(\mathbf{A})$	Condition number of $\mathbf{A}$. 44
$\mathcal{N}(\mathbf{A})$	The nullspace of $\mathbf{A}$. 7
$\text{rank}(\mathbf{A})$	The rank of $\mathbf{A}$. 124
i	The imaginary unit of a complex number. 117
λ^*	The eigenvalue at the root. 7
$\lambda^{(k)}$	The kth eigenvalue. 39
h	Mesh size. 49
$\mathbf{A}(\gamma)$	A parameter-dependent matrix $\mathbf{A}$. 1

CHAPTER 1

Introduction

In this thesis, we are interested in the numerical solution of some linear and nonlinear eigenvalue problems for real nonsymmetric n by n matrices. First, given a real parameter-dependent matrix $\mathbf{A}(\gamma)$, which is at least twice continuously differentiable with respect to γ, the problem is: as γ is varied, at what particular value of γ^* do two real or complex eigenvalues λ_1 and λ_2, say, of $\mathbf{A}(\gamma^*)$ coalesce at λ^* to form a 2-dimensional Jordan block? Second, we consider a related problem of computing the distance of a simple matrix to a nearby defective matrix. A defective matrix by definition, has a Jordan block of at least dimension two. Third, we study Newton's method for the computation of an algebraically simple complex eigenpair in which special attention is paid to the normalisation of the eigenvector.

Let us first consider a simple situation. Let $\mathbf{B}$ be an n by n symmetric matrix and $\mathbf{C}$ an n by n nonsymmetric matrix. Let γ be a real parameter and consider the following parameter-dependent eigenvalue problem

$$(\mathbf{B} + \gamma\mathbf{C})\mathbf{x} = \lambda\mathbf{x}; \quad or \quad \mathbf{A}(\gamma)\mathbf{x} = \lambda\mathbf{x}, \tag{1.1}$$

where $\mathbf{A}(\gamma) = (\mathbf{B} + \gamma\mathbf{C})$. When γ is zero, (1.1) becomes the standard symmetric eigenvalue problem of which, all the eigenvalues are real. However, as γ is increased monotonically from zero, the symmetric structure in $\mathbf{B}$ is lost because of the perturbation induced by the unsymmetric matrix $\gamma\mathbf{C}$ [29]. As a

result of this, for particular values of γ, two eigenvalues of $\mathbf{A}(\gamma)$ may coalesce to form a 2-dimensional Jordan block or they may not coalesce after all. It is easy to construct a 2 by 2 example where there is coalescence and another 2 by 2 example where coalescence does not occur. A particular case where coalescence is guaranteed to occur would be if $\mathbf{C}$ where skew-symmetric, so that as γ tends to infinity all the eigenvalues would tend to purely imaginary values. A physical example where coalescence does occur, is in the flutter problem (see, for example [53]) which we discuss next.

Flutter is a dynamic instability which can occur in structures in motion, subject to aerodynamic loading [7] as in the coalescence of two real eigenvalues in a supersonic panel flutter problem (see, for example [53]). The following parameter-dependent generalized eigenvalue problem

$$(\mathbf{K}_T + \gamma\mathbf{A})\mathbf{q} = \lambda\mathbf{M}\mathbf{q}, \tag{1.2}$$

arises from the finite element discretization of a supersonic panel flutter partial differential equation, where $\mathbf{K}_T$ and $\mathbf{M}$ are symmetric positive definite; the total stiffness and consistent mass matrices respectively, and $\mathbf{A}$ is the nonsymmetric aerodynamic load matrix. In this context, γ represents the dynamic pressure parameter and the pair $\mathbf{q}$ and λ represent displacements and eigenvalues respectively. When $\gamma = 0$, (1.2) corresponds to the symmetric eigenvalue problem of which all the eigenvalues are real and positive. However, as the dynamic pressure parameter γ is increased monotonically from zero, the first two smallest eigenvalues λ_1 and λ_2, say, move and coalesce together to λ^* at $\gamma = \gamma^*$ (which corresponds to the flutter speed [8, p. 423]) to form a 2-dimensional Jordan block and become complex conjugate eigenpairs when $\gamma > \gamma^*$ (see, for example, [43, pp. 2268-2269], [44, p. 748]). See more explanations in Section 2.3.

Another reason why the study of eigenvalue coalescence is important is because a knowledge of where they coalesce can be used to explain the stability of time-dependent ordinary or partial differential equations. For example, consider the following parameter-dependent ordinary differential equation

$$\frac{\mathrm{d}\mathbf{w}}{\mathrm{d}t} = -\mathbf{F}(\mathbf{w}, \gamma). \tag{1.3}$$

If we denote $\mathbf{w}_s$ as the steady state solution, then this means that at steady state

$$\frac{\mathrm{d}\mathbf{w}_s}{\mathrm{d}t} = -\mathbf{F}(\mathbf{w}_s, \gamma) = \mathbf{0}. \tag{1.4}$$

Now, if $\mathbf{w} = \mathbf{w}_s + \mathbf{u}$ is an approximation to the solution where $\mathbf{u}$ is a perturbing vector, then using Taylor Series (see, for example, [16, p. 18]), we can rewrite (1.3) as

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{w}_s + \mathbf{u}) = -\mathbf{F}(\mathbf{w}_s + \mathbf{u}, \gamma) = -\big[\mathbf{F}(\mathbf{w}_s, \gamma) + \mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)\mathbf{u} + h.o.t.\big],$$

where $\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)$ is the Jacobian with respect to $\mathbf{w}_s$ and γ. Using (1.4) and after neglecting second and higher order terms, we obtain

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = -\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)\mathbf{u}; \quad \text{or} \quad \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \mathbf{A}(\gamma)\mathbf{u}, \tag{1.5}$$

where $\mathbf{A}(\gamma) = -\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)$. Since the sign of the Jacobian above is negative, this means that the right half plane is stable. The behaviour of the solution of (1.5) depends on the spectrum of $\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)$, which is obtained by solving an eigenvalue problem of the form

$$\mathbf{A}(\gamma)\phi = \lambda\phi.$$

One possible scenario is when $\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)$ corresponds to a 2 by 2 matrix having repeated real eigenvalues λ^*, corresponding to a 2-dimensional Jordan block. Then the solution to the ordinary differential equation (1.5) can then be written as (see, for example, [45, 467-469])

$$\mathbf{u}(t) = (a_1 + a_2 t)e^{\lambda^* t}\phi + a_3 e^{\lambda^* t}\hat{\phi}, \tag{1.6}$$

for real constants a_1, a_2 and a_3, where $\big[-\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma) - \lambda^*\mathbf{I}\big]\hat{\phi} = \phi$, and $\hat{\phi}$ is a generalised eigenvector of $-\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)$ corresponding to λ^*. If the right half plane is stable, then the leftmost eigenvalues of the Jacobian $\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)$ determine the linearized stability of the steady state solutions of $\frac{\mathrm{d}\mathbf{w}}{\mathrm{d}t} = -\mathbf{F}(\mathbf{w}, \gamma)$ and one way of detecting Hopf bifurcation[1] points is to observe the first few

[1] [12, p. 61] A Hopf bifurcation occurs when a complex conjugate pair of eigenvalues of the parameter-dependent Jacobian $-\mathbf{F}_{\mathbf{w}_s}(\mathbf{w}_s, \gamma)$ crosses the imaginary axis. This is typical for one

leftmost eigenvalues. Hence, to recognise if instabilities are caused by real or complex eigenvalues crossing the imaginary axis, it is important to know when leftmost real eigenvalues coalesce and become leftmost complex conjugate eigenvalues (see, for example, Cliffe *et al.* [12, pp. 40, 99]).

An example that we discuss in detail with numerical results in Section 2.3 is the coalescence of two eigenvalues in a parameter-dependent nonsymmetric matrix to form a 2-dimensional Jordan block. This example arises from an eigenvalue problem that comes from the linearized stability of a partial differential equation. It is motivated by the computation of the stability of fluid flows governed by the steady-state Navier-Stokes equation as presented by Graham *et. al.*, in [28]. In our example, we seek the values of λ^* and γ^* such that two real leftmost eigenvalues of $\mathbf{A}(\gamma^*)$, obtained from the finite centred difference discritization of the resulting PDE eigenvalue problem, coalesce at λ^* to form a 2-dimensional Jordan block.

Another example of the importance of a 2-dimensional Jordan block is in the monitoring of the dynamics of power systems in electrical engineering (see, for example, Dodson *et. al.*, [19]). Here, Dodson *et. al.*, discussed the coalescence of two complex eigenvalues to form a 2-dimensional Jordan block as either power transfer or generator redispatch change. A numerical example arising from this application is given in Section 2.5.1 of Chapter 2. Two-dimensional Jordan blocks also arise when one considers the problem of computing the nearest defective matrix from a simple one. Alam & Bora [4] developed a numerical algorithm for computing the distance of a simple matrix from the set of matrices having a Jordan block of at least dimension two. A more detailed discussion of the history of this problem will follow in Section 1.2. In Chapter 3, we present an algorithm for the computation of a nearby defective matrix which is much more efficient than the Algorithm in Alam & Bora [4]. It is not guaranteed to find the nearest defective matrix since it is based on Newton's method. However, it succeeded in finding the nearest defective matrix in all the test examples.

Lastly, a further example of where Jordan blocks appear is in Freitag and Spence [22], who computed the distance of a stable matrix to the set of unstable matrices by computing a 2-dimensional Jordan block in a special class of

parameter-dependent problems of the form (1.5).

parameter-dependent Hamiltonian matrices.

The plan of this introductory chapter is as follows. In Section 1.1, we define some terms in use throughout this thesis. This will then be followed in Section 1.2 by a survey of previous attempts at finding an algorithm for computing a nearest defective matrix to a simple matrix. Next, Section 1.3 discusses two key mathematical tools used in this thesis: the ABCD Lemma and the implicit determinant method. Furthermore, we compare the implicit determinant method and inverse iteration, in Section 1.4. In Section 1.5, we discuss the theory of the Gauss-Newton method for solving over- and under-determined system of nonlinear equations. Section 1.6, presents a survey of Newton's method and inverse iteration with emphasises on the normalisation. Finally, in Section 1.7, we describe the structure of this thesis.

Next, we define some Linear Algebra terms and summarize the theory of Jordan blocks.

1.1 Background Theory: Jordan Blocks and some Important Definitions

In this section, we define some well known linear algebraic terms used throughout this thesis for quick reference and present background theory on Jordan blocks. Among other definitions, we define what a Jordan block is, algebraic and geometric multiplicities of the eigenvalue of a matrix, as well as what it means for a matrix to have a 2-dimensional Jordan block.

Let $\mathbf{A} \in \mathbb{R}$ be a real n by n matrix and $\lambda \in \mathbf{C}$ an eigenvalue of $\mathbf{A}$ corresponding to the nonzero eigenvector $\phi \in \mathbb{C}^n$, such that

$$\mathbf{A}\phi = \lambda\phi. \tag{1.7}$$

The vector ϕ is often referred to as a right eigenvector [40]. A left eigenvector corresponding to the eigenvalue λ is defined as any nonzero vector ψ that satisfies $\psi^T\mathbf{A} = \lambda\psi^T$. The term geometric multiplicity of an eigenvalue λ of $\mathbf{A}$ is defined as the dimension of the nullspace of $(\mathbf{A} - \lambda\mathbf{I})$. The algebraic multiplicity of an eigenvalue λ of $\mathbf{A}$ is its multiplicity as a root of the characteristic polynomial of $\mathbf{A}$ (see, for example, [60, p. 184]). We say λ is algebraically sim-

ple if it is a simple root of the characteristic polynomial. If λ is algebraically simple, then its corresponding left and right eigenvectors are not orthogonal, that is $\psi^T\phi \neq 0$ (see, for example, [21, p. 29, equation 2.2]).

Next, we define what it means for a matrix to have a 2-dimensional Jordan block. But before we do that, it is important to know what a Jordan block is first.

Definition 1.1.1. *[41, p. 358] A square upper-triangular matrix $\mathbf{J}(\lambda)$ that satisfies the following properties*

(a). all its main diagonal entries equal λ,

(b). all its entries on the first superdiagonal equal to one,

(c). all other entries are zero,

is called a Jordan block.

The following result explains the relationship between the Jordan decomposition of $\mathbf{A}$ and the Jordan block of $\mathbf{A}$.

Theorem 1.1.1. *[23, p. 317] If $\mathbf{A} \in \mathbb{C}^{n\times n}$, then there exists a nonsingular $\mathbf{Y} \in \mathbb{C}^{n\times n}$ such that*

$$\mathbf{J} = \mathbf{Y}^{-1}\mathbf{A}\mathbf{Y} = \operatorname{diag}\big(\mathbf{J}(\lambda_1), \mathbf{J}(\lambda_2), \ldots, \mathbf{J}(\lambda_t)\big),$$

where

$$\mathbf{J}(\lambda_i) = \begin{bmatrix} \lambda_i & 1 & & \cdots & 0 \\ 0 & \lambda_i & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \cdots & & 0 & \lambda_i \end{bmatrix},$$

is an $m(\lambda_i) \times m(\lambda_i)$ matrix and $m(\lambda_1) + m(\lambda_2) + \cdots + m(\lambda_t) = n$, $m(\lambda_i)$ is the algebraic multiplicity of λ_i and t is the number of linearly independent eigenvectors of $\mathbf{A}$ corresponding to the number of blocks.

A way to recognise if the matrix $\mathbf{A}(\gamma)$ has a 2-dimensional Jordan block is given in the next definition.

Definition 1.1.2. *[22] Let λ^* be an eigenvalue of $\mathbf{A}(\gamma^*)$. $\mathbf{A}(\gamma^*)$ has a 2-dimensional Jordan block corresponding to the eigenvalue λ^* if λ^* has algebraic multiplicity 2 and geometric multiplicity 1.*

An immediate consequence of λ^* being algebraically double and geometrically simple in the above definition is explained as follows. If
$\boldsymbol{\phi}^* \in \mathcal{N}\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\backslash\{\mathbf{0}\}$ and $\boldsymbol{\psi}^* \in \mathcal{N}\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)^T\backslash\{\mathbf{0}\}$, then

$$\boldsymbol{\psi}^{*T}\boldsymbol{\phi}^* = 0, \tag{1.8}$$

and there exists a generalised eigenvector $\hat{\boldsymbol{\phi}}^*$ corresponding to λ^* which satisfies

$$\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\hat{\boldsymbol{\phi}}^* = \boldsymbol{\phi}^*, \quad \textit{and} \qquad \boldsymbol{\psi}^{*T}\hat{\boldsymbol{\phi}}^* \neq 0. \tag{1.9}$$

We have used the Jordan chain equations (see, for example [41, pp. 359]) to arrive at the last equation and the condition $\boldsymbol{\psi}^{*T}\hat{\boldsymbol{\phi}}^* \neq 0$ ensures that the dimension of the Jordan block is exactly 2. After premultiplying both sides of (1.9) by $\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)$, we obtain

$$\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)^2\hat{\boldsymbol{\phi}}^* = \big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\boldsymbol{\phi}^* = \mathbf{0}.$$

This shows that the algebraic multiplicity of λ^* is at least two and that $\hat{\boldsymbol{\phi}}^*$ is indeed a generalised eigenvector.

Before we continue, we give some further definitions in use.

Definition 1.1.3. *[4] An $n \times n$ matrix is **simple** if it has n distinct eigenvalues.*

Definition 1.1.4. *[60, p. 185] An eigenvalue is said to be **defective** if its algebraic multiplicity is greater than its geometric multiplicity. A matrix is said to be defective if it has one or more defective eigenvalues.*

Definition 1.1.5. *[3, p. 367] The distance $\mathrm{d}(\mathbf{A})$ of a simple matrix $\mathbf{A}$ from a nearby defective one, $\mathbf{B}$ is defined as,*

$$\mathrm{d}(\mathbf{A}) = \inf\{\|\mathbf{A} - \mathbf{B}\| : \mathbf{B} \ \textit{is defective}\}, \tag{1.10}$$

and

$$\mathrm{gap}(\mathbf{A}) = \min_{i \neq j} \frac{|\lambda_i - \lambda_j|}{2}, \tag{1.11}$$

where the λ_i's for $i = 1, 2, \ldots n$, are the eigenvalues of $\mathbf{A}$.

In the next section, we give a survey of previous attempts at finding a nearest defective matrix to a simple matrix.

1.2 Computing a Nearby Defective Matrix

As mentioned in the introductory section, one of the applications where a 2-dimensional Jordan block arises is in the computation of a nearest defective matrix from a simple one. Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be a simple matrix. The next problem that we seek an answer to in this thesis, is to describe how to find a nearby defective matrix to $\mathbf{A}$. In more precise terms, we attempt to provide a partial answer to a question posed by Wilkinson (see, [62, pp.90-93]) *i.e.*, "Given a simple matrix $\mathbf{A}$, find $\mathrm{d}(\mathbf{A})$ and a defective matrix $\mathbf{B}$ such that $\mathrm{d}(\mathbf{A}) = \inf\{\|\mathbf{A} - \mathbf{B}\| : \mathbf{B} \text{ is defective}\}$." The main reason for computing a nearby defective matrix to $\mathbf{A}$ is because: if $\mathbf{A}$ has a nearby defective matrix, then it has ill-conditioned eigenvalues [64]. The focus in this section, is to present in a chronological order, a brief survey of previous attempts at finding a nearest defective matrix from a simple one. We survey the contributions of Ruhe, Wilkinson, Malyshev and Alam & Bora.

In an attempt to provide an answer to Wilkinson's problem, Ruhe [50, p. 58] gave a bound for the distance between a matrix $\mathbf{A}$ having distinct eigenvalues and the set of matrices having at least two coinciding eigenvalues, in terms of the angle between a vector and a subspace.

In Wilkinson's paper [64], he assumes that if $\mathbf{A}$ is a matrix with an ill-conditioned eigenvalue z and $\mathbf{B} = \mathbf{Q}\mathbf{A}\mathbf{Q}^H$, $\mathbf{Q}$ unitary, then there exists a nearby matrix $\mathbf{B} + \mathbf{E}$ having multiple eigenvalues. In that paper, he gave a sharper bound for the distance between $\mathbf{A}$ and the set of matrices having multiple eigenvalues than Ruhe's [50]. Wilkinson's proof uses the inner product $s = \mathbf{y}^H\mathbf{x}$, where $\mathbf{y}$ and $\mathbf{x}$ are the unit left and right eigenvectors corresponding to the eigenvalue z. The reciprocal of s is the condition number of a simple eigenvalue of $\mathbf{A}$. Thus, a small s implies that the condition number of the eigenvalue is large. He shows that if s is 'small', then there exists a perturbed matrix $\mathbf{B} + \mathbf{E}$ having z as a multiple eigenvalue and $\mathbf{A} + \mathbf{F} = \mathbf{Q}^H\mathbf{B}\mathbf{Q} + \mathbf{Q}^H\mathbf{E}\mathbf{Q}$, $\mathbf{F} = \mathbf{Q}^H\mathbf{E}\mathbf{Q}$ such that the ratio $\|\mathbf{F}\|/\|\mathbf{A}\|$, is small [64].

More recently, Malyshev [37], proved that the 2-norm distance from an $n \times n$ matrix $\mathbf{A}$ to the set of matrices with multiple eigenvalues $z \in \mathbb{C}$ and $\omega \in \mathbb{R}$ is given by

$$\mathrm{d}(\mathbf{A}) = \min_{z\in\mathbf{C}} \max_{\omega\geq 0} \sigma_{2n-1} \begin{bmatrix} \mathbf{A} - z\mathbf{I} & \omega\mathbf{I} \\ 0 & \mathbf{A} - z\mathbf{I} \end{bmatrix},$$

where σ_j is the jth singular value. However, as Malyshev admits, the above expression for $\mathrm{d}(\mathbf{A})$ is mainly of theoretical interest and not so useful as a computational result. This is because the outer minimization is a hard optimisation problem.

If $\mathbf{A}$ is normal, Alam [3], gives a procedure for constructing the nearest defective matrix to $\mathbf{A}$. His construction is based on an appropriate pair of eigenvalues of $\mathbf{A}$ and their corresponding unit eigenvectors. The matrix $\mathbf{B}$ was constructed such that $\mathrm{d}(\mathbf{A}) = \|\mathbf{A} - \mathbf{B}\|$. Alam's paper [3], consists of two important results: the first result is important because it provides a formula for finding the nearest defective matrix from a normal matrix $\mathbf{A}$. Moreover, the formula is based on the normalized left and right singular vectors $\mathbf{u}$ and $\mathbf{v}$ of $\mathbf{A} - z\mathbf{I}$ corresponding to its smallest singular value $\sigma_n \neq 0$ such that $\mathbf{u}^H\mathbf{v} = 0$, and z is a defective eigenvalue of $\mathbf{B}$ with $\sigma_n = \|\mathbf{A} - \mathbf{B}\|$. The second result is important because, it tells us how to find z *i*.e., the midpoint of a pair (λ_i, λ_j) of eigenvalues of $\mathbf{A}$ such that $|\lambda_i - \lambda_j| = 2\,\mathrm{gap}(\mathbf{A})$, where $\mathrm{gap}(\mathbf{A})$ is as defined as in (1.11)

$$z = \frac{\lambda_i + \lambda_j}{2}. \tag{1.12}$$

With this value of z, Alam and Bora constructed two defective matrices $\mathbf{B}, \mathbf{B}'$ by the formulae [3]

$$\mathbf{B} = \mathbf{A} - \frac{(\lambda_i - \lambda_j)}{2} \frac{(\mathbf{x}_i - \mathbf{x}_j)}{\sqrt{2}} \frac{(\mathbf{x}_i + \mathbf{x}_j)^H}{\sqrt{2}} = \mathbf{A} - \frac{1}{4}(\lambda_i - \lambda_j)(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i + \mathbf{x}_j)^H, \tag{1.13}$$

and

$$\mathbf{B}' = \mathbf{A} - \frac{(\lambda_i - \lambda_j)}{2} \frac{(\mathbf{x}_i + \mathbf{x}_j)}{\sqrt{2}} \frac{(\mathbf{x}_i - \mathbf{x}_j)^H}{\sqrt{2}} = \mathbf{A} - \frac{1}{4}(\lambda_i - \lambda_j)(\mathbf{x}_i + \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^H, \tag{1.14}$$

such that $\mathrm{d}(\mathbf{A}) = \mathrm{gap}(\mathbf{A}) = \|\mathbf{A} - \mathbf{B}\| = \|\mathbf{A} - \mathbf{B}'\|$. Here, λ_i and $\mathbf{x}_i$ for $i = 1, \ldots, n$ are n distinct eigenvalues of $\mathbf{A}$ and their corresponding unit eigen-

vectors respectively.

When $\mathbf{A}$ is nonnormal, Alam and Bora [4, p. 292], presented an algorithm for finding the nearest defective matrix to a simple matrix, and the distance between them. Alam and Bora [4, p. 284] proved that given a complex n by n matrix, with $z \in \mathbb{C}\backslash\Lambda(\mathbf{A})$ which has to be found, $\Lambda(\mathbf{A})$ is the spectrum of $\mathbf{A}$, $\mathbf{u}$ and $\mathbf{v}$ are a pair of normalized left and right singular vectors of $\mathbf{A} - z\mathbf{I}$ corresponding to the smallest singular value ε such that $\mathbf{u}^H\mathbf{v} = 0$, then the nearest defective matrix to $\mathbf{A}$ is given by the formula; $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$. It was shown that $\mathbf{u}$ and $\mathbf{v}$ are left and right eigenvectors of $\mathbf{B}$ corresponding to the eigenvalue z *i*.e., $\mathbf{u}^H\mathbf{B} = z\mathbf{u}^H$ and $\mathbf{B}\mathbf{v} = z\mathbf{v}$. Since $\mathbf{u}^H\mathbf{v} = 0$, this implies that z is a multiple eigenvalue of $\mathbf{B}$, hence $\mathrm{d}(\mathbf{A}) = \varepsilon$. However, their algorithm for finding the values of z and ε, and the nearest defective matrix to $\mathbf{A}$ relies on the computation of the ε-pseudospectrum of $\mathbf{A}$ which we now describe.

The ε-pseudospectra $\Lambda_\varepsilon(\mathbf{A})$ of a matrix $\mathbf{A}$, can be defined as [61, p. 458]

$$\Lambda_\varepsilon(\mathbf{A}) = \bigcup_{\mathbf{B}\in\mathbb{A}(\varepsilon)} \Lambda(\mathbf{B}), \tag{1.15}$$

where

$$\mathbb{A}(\varepsilon) = \{\mathbf{B} \in \mathbb{C}^{n\times n} : \|\mathbf{A} - \mathbf{B}\| \leq \varepsilon\}.$$

For any $\varepsilon > 0$, the ε-pseudospectrum of $\mathbf{A}$, $\Lambda_\varepsilon(\mathbf{A})$ consists of nontrivial components and the interior of each of its component contains at least one eigenvalue of $\mathbf{A}$. As ε is increased, the components of $\Lambda_\varepsilon(\mathbf{A})$ coalesce and z, the eigenvalue of the defective matrix $\mathbf{B}$ is found from the point of coalescence. This notion of ε-pseudospectra was used to show that if z is a point of coalescence of two components of $\Lambda_\varepsilon(\mathbf{A})$, then z is a multiple eigenvalue of the defective matrix $\mathbf{B}$ such that $\varepsilon = \|\mathbf{A} - \mathbf{B}\|$. Though the paper [4], provides the solution to Wilkinson's problem, the algorithm given for computing a nearest defective matrix is slow and impractical for large matrices. This is because it requires the computation of the ε-pseudospectrum of $\mathbf{A}$ and a decision as to when two components of $\Lambda_\varepsilon(\mathbf{A})$ coalesce is needed, and it is not obvious how this may be achieved automatically. The first drawback has been circumvented with the development of a new free software `eigtool` by Wright [67]. An excellent overview of previous methods for computing the nearest defective matrix has

been given by Overton [47].

In the next section, we present a review on the implicit determinant method of Spence and Poulton for the solution of a nonlinear eigenvalue problem arising from photonic crystals [55] as well as Keller's [33] ABCD Lemma.

1.3 Background: ABCD Lemma and the Implicit Determinant Method

In this section, we present two key mathematical tools that will be of great use in this thesis. In the first case, we present Keller's [33] ABCD Lemma. Secondly, we review the implicit determinant method of Spence and Poulton [55] which makes use of a special case of the ABCD Lemma and Cramer's rule. The key results in this section are Lemmas 1.3.1 and 1.3.2.

First, we present the one-dimensional version of Keller's [33] ABCD Lemma.

Lemma 1.3.1. *The "ABCD" Lemma*
Let $\mathbf{A}$ *be an* n *by* n *matrix,* $\mathbf{b}, \mathbf{c} \in \mathbb{R}^n$ *and* $d \in \mathbb{R}$. *Let*

$$\mathbf{M} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix}, \tag{1.16}$$

be an $(n+1)$ *by* $(n+1)$ *real matrix.*

(a). Suppose that $\mathbf{A}$ *is nonsingular, then there exists the following decomposition of* $\mathbf{M}$,

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{c}^T\mathbf{A}^{-1} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0}^T & d - \mathbf{c}^T\mathbf{A}^{-1}\mathbf{b} \end{bmatrix}. \tag{1.17}$$

The matrix $\mathbf{M}$ *is nonsingular if and only if* $d - \mathbf{c}^T\mathbf{A}^{-1}\mathbf{b} \neq 0$.

(b). If $\mathbf{A}$ *is singular of* $\text{rank}(\mathbf{A}) = n - 1$, *then* $\mathbf{M}$ *is nonsingular if and only if* $\boldsymbol{\psi}^T\mathbf{b} \neq 0$, *for all* $\boldsymbol{\psi} \in \mathcal{N}(\mathbf{A}^T)\backslash\{\mathbf{0}\}$ *and* $\mathbf{c}^T\boldsymbol{\phi} \neq 0$, *for all* $\boldsymbol{\phi} \in \mathcal{N}(\mathbf{A})\backslash\{\mathbf{0}\}$.

Proof: See [33]. ∎

Next, we describe Spence and Poulton's implicit determinant method as formulated in [55]. The aim of presenting the implicit determinant method is be-

cause we want to extend it to the parameter-dependent nonsymmetric matrix case to find a 2-dimensional Jordan block.

The implicit determinant method of Spence and Poulton [55] is a method of converting a problem for $n \times n$ matrices into an equivalent scalar problem. We can solve the scalar problem in a number of ways, for example, using the bisection method. The fact that it is efficient to implement Newton's method is an added advantage. In the paper [55], the theory of the implicit determinant method was given for the case in which $\mathbf{A}(\gamma)$ is Hermitian, and comparisons were made on the convergence of the implicit determinant method and nonlinear inverse iteration applied to a nonlinear eigenvalue problem arising in a photonic crystal problem.

Given a parameter-dependent Hermitian matrix $\mathbf{A}(\gamma)$ and assume $\mathbf{A}(\gamma)$ is a smooth function of γ. Let [55, p. 69]

$$\mathbf{A}(\gamma)\mathbf{x} = \mathbf{0}, \quad \text{where} \quad \mathbf{x} \neq \mathbf{0}, \tag{1.18}$$

be a parameter-dependent eigenvalue problem.

Consider the following $(n+1)$ by $(n+1)$ bordered linear system of equations [55, p. 70],

$$\begin{bmatrix} \mathbf{A}(\gamma) & \mathbf{b} \\ \mathbf{b}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}, \tag{1.19}$$

which shows that the eigenvector $\mathbf{x}$ is normalised using $\mathbf{b}^H\mathbf{x} = 1$. The following result is the main mathematical tool of Spence and Poulton's implicit determinant method.

Lemma 1.3.2. *[55, pp. 70] Let* $(\mathbf{x}^*, \gamma^*)$ *solve* (1.18) *with* $\mathbf{A}(\gamma)$ *Hermitian. Assume that zero is a simple eigenvalue of* $\mathbf{A}(\gamma^*)$*, such that*

(a). $\dim \mathcal{N}\left[\mathbf{A}(\gamma^*)\right] = 1.$

(b). For some $\mathbf{b} \in \mathbb{C}^n \backslash \{\mathbf{0}\}$*, assume*

$$\mathbf{b}^H\mathbf{x}^* \neq 0. \tag{1.20}$$

Then the $(n+1)$ *by* $(n+1)$ *matrix* $\mathbf{M}(\gamma)$ *defined by*

$$\mathbf{M}(\gamma) = \begin{bmatrix} \mathbf{A}(\gamma) & \mathbf{b} \\ \mathbf{b}^H & 0 \end{bmatrix},$$

is nonsingular at $\gamma = \gamma^*$.

Proof: See [33]. ∎

From the result of Lemma 1.3.2, $\mathbf{M}(\gamma)$ is nonsingular at the root. Following [55], this means that by an application of the implicit function theorem (see, for example, [56, p. 186]) $\mathbf{M}(\gamma)$ is nonsingular for γ near γ^* because $\mathbf{A}(\gamma)$ is a smooth function of γ. Therefore, from (1.19) $\mathbf{x}$ and f are smooth functions of γ and we can write $\mathbf{x} = \mathbf{x}(\gamma)$ and $f = f(\gamma)$. So that (1.19) becomes

$$\begin{bmatrix} \mathbf{A}(\gamma) & \mathbf{b} \\ \mathbf{b}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(\gamma) \\ f(\gamma) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \tag{1.21}$$

Now, by applying Cramer's rule (see [32, p. 414]) to (1.21), we obtain

$$f(\gamma) = \frac{\det \mathbf{A}(\gamma)}{\det \mathbf{M}(\gamma)}. \tag{1.22}$$

As stated in [55], because $\mathbf{A}(\gamma)$ and $\mathbf{M}(\gamma)$ are both Hermitian, this means that $f(\gamma)$ is real. We conclude by saying that the main idea behind the implicit determinant method is that if $\mathbf{M}(\gamma)$ is nonsingular, then $f(\gamma) = 0$ if and only if $\mathbf{A}(\gamma)$ is singular. So we seek zeros of $f(\gamma)$ as a way of finding the zeros of the determinant of $\mathbf{A}(\gamma)$. Spence and Poulton continue by finding the solution of $f(\gamma) = 0$ using Newton's method, which requires the calculation of $f_\gamma(\gamma)$, where $f_\gamma(\gamma) = \frac{\mathrm{d}}{\mathrm{d}\gamma} f(\gamma)$. This is accomplished by solving

$$\begin{bmatrix} \mathbf{A}(\gamma) & \mathbf{b} \\ \mathbf{b}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\gamma(\gamma) \\ f_\gamma(\gamma) \end{bmatrix} = - \begin{bmatrix} \mathbf{A}'(\gamma)\mathbf{x}(\gamma) \\ 0 \end{bmatrix},$$

obtained by differentiating both sides of (1.21) with respect to γ. After which the sequence of γ iterates is computed by $\gamma^{(k+1)} = \gamma^{(k)} - f(\gamma^{(k)})/f_\gamma(\gamma^{(k)})$, for $k = 0, 1, 2, \ldots$. Using the above matrix equation, $f_\gamma(\gamma^*)$ was shown to be equal to $-\mathbf{x}^{*H}\mathbf{A}'(\gamma^*)\mathbf{x}(\gamma^*)$. Hence, $f_\gamma(\gamma^*)$ is nonzero provided $\mathbf{x}^{*H}\mathbf{A}'(\gamma^*)\mathbf{x}(\gamma^*)$ is

nonzero.

Note that, Freitag and Spence in [22], extended the method recounted above to a special class of parameter-dependent Hamiltonian matrices by computing a 2-dimensional Jordan block to solve a distance to instablility problem. In this case, $\mathbf{A}(\gamma)$ is $\mathbf{H}(\gamma) - i\omega\mathbf{I}$ where $\mathbf{H}(\gamma)$ is the parameter-dependent Hamiltonian, $\omega \in \mathbb{R}$ and $\mathbf{M}(\gamma) = \mathbf{M}(\gamma, \omega)$ is now nonsymmetric-depending on two parameters γ and ω. In Chapter 2 of this thesis, we extend the idea of Spence and Poulton's implicit determinant method [55, p. 71] further, to the case of computing a 2-dimensional Jordan block from a parameter-dependent nonsymmetric matrix. This version of the implicit determinant method shows that there is a relationship between the zeros of $f(\lambda, \gamma)$ and the determinant of $(\mathbf{A}(\gamma) - \lambda\mathbf{I})$.

First and foremost, in the next section, we present the implicit determinant method for a nonsymmetric matrix and compare it with inverse iteration.

1.4 A Comparison of the Implicit Determinant Method and Inverse Iteration

Let $\mathbf{A}$ be a real n by n nonsymmetric matrix. In this section, we give the nonsymmetric version of inverse iteration and then extend the implicit determinant method of Spence and Poulton to a nonsymmetric $\mathbf{A}$. We conclude by comparing this version of the implicit determinant method with inverse iteration. The discussion on inverse iteration in this section is a special case of [21] for the standard eigenvalue problem.

Recall from (1.7) that $(\mathbf{A} - \lambda\mathbf{I})\phi = \mathbf{0}$. So, if we add to (1.7) the eigenvector normalization $\mathbf{c}^T\phi = 1$, then the extended system of nonlinear equations becomes: (see, also [21, p. 29])

$$\mathbf{F}(\mathbf{w}) = \begin{bmatrix} (\mathbf{A} - \lambda\mathbf{I})\phi \\ \mathbf{c}^T\phi - 1 \end{bmatrix} = \mathbf{0}, \tag{1.23}$$

where $\mathbf{w} = [\phi^T, \lambda]$. Using the ABCD Lemma [33], it can be shown that the

Jacobian $\mathbf{F}_{\mathbf{w}}(\mathbf{w})$ is nonsingular,

$$\mathbf{F}_{\mathbf{w}}(\mathbf{w}) = \begin{bmatrix} \mathbf{A} - \lambda\mathbf{I} & -\boldsymbol{\phi} \\ \mathbf{c}^T & 0 \end{bmatrix}. \tag{1.24}$$

at the root. Hence, its inverse exists at an algebraically simple eigenvalue (*i.e.*, $\boldsymbol{\psi}^T\boldsymbol{\phi} \neq 0$, for all $\boldsymbol{\psi} \in \mathcal{N}(\mathbf{A}^T - \lambda\mathbf{I})\backslash\{\mathbf{0}\}$ and $\boldsymbol{\phi} \in \mathcal{N}(\mathbf{A} - \lambda\mathbf{I})\backslash\{\mathbf{0}\}$) and if $\mathbf{c}$ is chosen such that $\mathbf{c}^T\boldsymbol{\phi} \neq 0$. Newton's method

$$\begin{aligned} \mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\Delta\mathbf{w}^{(k)} &= -\mathbf{F}(\mathbf{w}^{(k)}) \\ \mathbf{w}^{(k+1)} &= \mathbf{w}^{(k)} + \Delta\mathbf{w}^{(k)}, \end{aligned} \tag{1.25}$$

with $\mathbf{c}^T\boldsymbol{\phi}^{(k)} = 1$, now becomes

$$\begin{bmatrix} \mathbf{A} - \lambda^{(k)}\mathbf{I} & -\boldsymbol{\phi}^{(k)} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\phi}^{(k)} \\ \Delta\lambda^{(k)} \end{bmatrix} = -\begin{bmatrix} (\mathbf{A} - \lambda^{(k)}\mathbf{I})\boldsymbol{\phi}^{(k)} \\ \mathbf{c}^T\boldsymbol{\phi}^{(k)} - 1 \end{bmatrix}.$$

By expanding the above, we have the following system of equations

$$\begin{aligned} (\mathbf{A} - \lambda^{(k)}\mathbf{I})\Delta\boldsymbol{\phi}^{(k)} - \Delta\lambda^{(k)}\boldsymbol{\phi}^{(k)} &= -(\mathbf{A} - \lambda^{(k)}\mathbf{I})\boldsymbol{\phi}^{(k)} \\ \mathbf{c}^T\Delta\boldsymbol{\phi}^{(k)} &= 0. \end{aligned}$$

After collecting like terms in the first equation above and using the relation $\boldsymbol{\phi}^{(k+1)} = \boldsymbol{\phi}^{(k)} + \Delta\boldsymbol{\phi}^{(k)}$, we obtain

$$(\mathbf{A} - \lambda^{(k)}\mathbf{I})\boldsymbol{\phi}^{(k+1)} = \Delta\lambda^{(k)}\boldsymbol{\phi}^{(k)} \tag{1.26}$$

Upon division of both sides by $\Delta\lambda^{(k)}$ and letting $\mathbf{w}^{(k)} = \frac{\boldsymbol{\phi}^{(k+1)}}{\Delta\lambda^{(k)}}$ we have

$$(\mathbf{A} - \lambda^{(k)}\mathbf{I})\mathbf{w}^{(k)} = \boldsymbol{\phi}^{(k)}, \tag{1.27}$$

using the fact that $\mathbf{c}^T\Delta\boldsymbol{\phi}^{(k)} = 0$, we have $\mathbf{c}^T\boldsymbol{\phi}^{(k+1)} = \mathbf{c}^T(\boldsymbol{\phi}^{(k)} + \Delta\boldsymbol{\phi}^{(k)}) = 1$.

Hence, $\mathbf{c}^T\mathbf{w}^{(k)} = \frac{1}{\Delta\lambda^{(k)}}$, from which $\Delta\lambda^{(k)} = \frac{1}{\mathbf{c}^T\mathbf{w}^{(k)}}$. Therefore,

$$\begin{aligned}\boldsymbol{\phi}^{(k+1)} &= \Delta\lambda^{(k)}\mathbf{w}^{(k)} \\ &= \frac{\mathbf{w}^{(k)}}{\mathbf{c}^T\mathbf{w}^{(k)}}.\end{aligned} \tag{1.28}$$

By making use of (1.25) we have

$$\begin{aligned}\lambda^{(k+1)} &= \lambda^{(k)} + \Delta\lambda^{(k)} \\ &= \lambda^{(k)} + \frac{1}{\mathbf{c}^T\mathbf{w}^{(k)}}.\end{aligned} \tag{1.29}$$

From the above analysis, Algorithm 1 is immediate.

Algorithm 1 Inverse Iteration and Newton's Method

Input: $\boldsymbol{\phi}^{(0)}$, $\lambda^{(0)}$, $\mathbf{c}^{(0)}$ such that $\mathbf{c}^T\boldsymbol{\phi}^{(0)} = 1$, tol.

1: **for** $k = 1, 2, \cdots$, until convergence **do**
2: Solve $(\mathbf{A} - \lambda^{(k)}\mathbf{I})\mathbf{w}^{(k)} = \boldsymbol{\phi}^{(k-1)}$.
3: Compute $\Delta\lambda^{(k)} = \frac{1}{\mathbf{c}^T\mathbf{w}^{(k)}}$.
4: Compute $\lambda^{(k+1)} = \lambda^{(k)} + \Delta\lambda^{(k)}$.
5: Update $\boldsymbol{\phi}^{(k+1)} = \Delta\lambda^{(k)}\mathbf{w}^{(k)}$.
6: Test for convergence.
7: **end for**

Output: $\boldsymbol{\phi}^*$ and λ^*.

Next, we describe the implicit determinant method for a nonsymmetric $\mathbf{A}$. Consider the following $(n+1)$ by $(n+1)$ bordered linear system of equations [55, p. 70],

$$\begin{bmatrix} (\mathbf{A} - \lambda\mathbf{I}) & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \tag{1.30}$$

Lemma 1.4.1. *Let* $(\mathbf{x}^*, \lambda^*)$ *solve* (1.30). *Assume that zero is a simple eigenvalue of* $(\mathbf{A} - \lambda^*\mathbf{I})$, *such that*

(a). $\dim \mathcal{N}\left[(\mathbf{A} - \lambda^*\mathbf{I})\right] = 1.$

(b). For some $\mathbf{b},\ \mathbf{c} \in \mathbb{R}^n \backslash \{\mathbf{0}\}$, *assume*

$$\boldsymbol{\psi}^{*T}\mathbf{b} \neq 0, \quad \textit{and} \quad \mathbf{c}^T\mathbf{x}^* \neq 0, \tag{1.31}$$

for all $\boldsymbol{\psi}^* \in N\big[(\mathbf{A}^T - \lambda^*\mathbf{I})\big]$.

Then the $(n+1)$ *by* $(n+1)$ *matrix* $\mathbf{M}(\lambda^*)$ *defined by*

$$\mathbf{M}(\lambda^*) = \begin{bmatrix} (\mathbf{A} - \lambda^*\mathbf{I}) & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix},$$

is nonsingular.

Proof: See [33]. ∎

Since the result of Lemma 1.4.1 shows that $\mathbf{M}(\lambda^*)$ is nonsingular, then following [55], this means that by an application of the implicit function theorem (see, for example, [56, p. 186]), $\mathbf{M}(\lambda)$ is nonsingular for λ near λ^* because $(\mathbf{A} - \lambda\mathbf{I})$ is a smooth function of λ. Therefore, from (1.30) $\mathbf{x}$ and f are smooth functions of λ and we can write $\mathbf{x} = \mathbf{x}(\lambda)$ and $f = f(\lambda)$. So that (1.30) becomes

$$\begin{bmatrix} (\mathbf{A} - \lambda\mathbf{I}) & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(\lambda) \\ f(\lambda) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \tag{1.32}$$

Now, by applying Cramer's rule (see [32, p. 414]) to (1.32), we obtain

$$f(\lambda) = \frac{\det(\mathbf{A} - \lambda\mathbf{I})}{\det \mathbf{M}(\lambda)}. \tag{1.33}$$

By the implicit determinant method, if $\mathbf{M}(\lambda)$ is nonsingular, then $f(\lambda) = 0$ if and only if $(\mathbf{A} - \lambda\mathbf{I})$ is singular, which is attainable at the root. So we seek zeros of $f(\lambda)$ as a way of finding the zeros of the determinant of $(\mathbf{A} - \lambda\mathbf{I})$. To find the solution of $f(\lambda) = 0$ using Newton's method, we need $f_\lambda(\lambda)$, where $f_\lambda(\lambda) = \frac{\mathrm{d}}{\mathrm{d}\lambda} f(\lambda)$. This means we have to differentiate (1.32) with respect to λ and solve

$$\begin{bmatrix} (\mathbf{A} - \lambda\mathbf{I}) & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\lambda(\lambda) \\ f_\lambda(\lambda) \end{bmatrix} = \begin{bmatrix} \mathbf{x}(\lambda) \\ 0 \end{bmatrix}. \tag{1.34}$$

After which the sequence of λ iterates is computed by

$$\lambda^{(k+1)} = \lambda^{(k)} - \frac{f(\lambda^{(k)})}{f_\lambda(\lambda^{(k)})}, \tag{1.35}$$

for $k = 0, 1, 2, \ldots$, until convergence. At the root, observe that by expanding

the first row of (1.34), one obtains

$$(\mathbf{A} - \lambda^*\mathbf{I})\mathbf{x}_\lambda(\lambda^*) + f_\lambda(\lambda^*)\mathbf{b} = \mathbf{x}(\lambda^*). \tag{1.36}$$

Hence, after premultiplying both sides by ${\psi^*}^T$, then

$$f_\lambda(\lambda^*) = \frac{{\psi^*}^T\mathbf{x}(\lambda^*)}{{\psi^*}^T\mathbf{b}}, \quad since \quad {\psi^*}^T\mathbf{b} \neq 0. \tag{1.37}$$

But for an algebraically simple eigenvalue, the left and right eigenvector are not orthogonal *i.e.*, ${\psi^*}^T\mathbf{x}(\lambda^*) \neq 0$. Therefore,

$$f_\lambda(\lambda^*) \neq 0. \tag{1.38}$$

Algorithm 2 is now immediate.

Algorithm 2 Implicit Determinant Method Algorithm for a Simple Matrix

Input: Choose $\mathbf{b}, \mathbf{c}$, $\lambda^{(0)}$, such that $\mathbf{M}(\lambda^{(0)})$ is nonsingular, tol.

1: **for** $k = 1, 2, \cdots$, until convergence **do**
2: Solve (1.32) for $\mathbf{x}(\lambda)$ and $f(\lambda)$.
3: Solve (1.34) for $\mathbf{x}_\lambda(\lambda)$ and $f_\lambda(\lambda)$.
4: Update

$$\lambda^{(k+1)} = \lambda^{(k)} - \frac{f(\lambda^{(k)})}{f_\lambda(\lambda^{(k)})}.$$

5: Test for convergence.
6: **end for**

Output: $\mathbf{x}(\lambda^*)$ and λ^*.

Stop Algorithm 2 as soon as

$$\|f(\lambda^{(k)})\| \leq tol.$$

Now, we present the theory to explain the link between the implicit determinant method and inverse iteration. For ease of notation, we shall drop the superscripts k and write $\lambda^{(k+1)} = \lambda^+$ and $\lambda^{(k)} = \lambda$.

We start by assuming that $\lambda \neq \lambda^*$, which implies $(\mathbf{A} - \lambda\mathbf{I})$ is nonsingular.

Observe by expanding along the first row of (1.32), that

$$(\mathbf{A}-\lambda\mathbf{I})\mathbf{x}(\lambda)+\mathbf{b}f(\lambda)=\mathbf{0}, \quad \text{and} \quad \mathbf{x}(\lambda)+(\mathbf{A}-\lambda\mathbf{I})^{-1}\mathbf{b}f(\lambda)=\mathbf{0}.$$

Premultiply both sides by $\mathbf{c}^T$ and solve for $f(\lambda)$ using the second row of (1.32) to obtain

$$f(\lambda)=-\frac{1}{\mathbf{c}^T(\mathbf{A}-\lambda\mathbf{I})^{-1}\mathbf{b}}. \tag{1.39}$$

Similarly, it can be shown by using the first row of (1.34) and $\mathbf{c}^T\mathbf{x}_\lambda(\lambda)$ from the second row that

$$f_\lambda(\lambda)=\frac{\mathbf{c}^T(\mathbf{A}-\lambda\mathbf{I})^{-1}\mathbf{x}(\lambda)}{\mathbf{c}^T(\mathbf{A}-\lambda\mathbf{I})^{-1}\mathbf{b}}. \tag{1.40}$$

Note from (1.27), that if we replace $\mathbf{w}$ with $\mathbf{y}$ and ϕ with $\mathbf{x}$, that is,

$$(\mathbf{A}-\lambda\mathbf{I})\mathbf{y}=\mathbf{x}, \quad \text{then} \quad \mathbf{y}=(\mathbf{A}-\lambda\mathbf{I})^{-1}\mathbf{x}(\lambda),$$

and we can rewrite (1.40) as

$$f_\lambda(\lambda)=\frac{\mathbf{c}^T\mathbf{y}}{\mathbf{c}^T(\mathbf{A}-\lambda\mathbf{I})^{-1}\mathbf{b}}.$$

Now, it is easy to see that (1.35) reduces to

$$\lambda^+=\lambda+\frac{1}{\mathbf{c}^T\mathbf{y}}.$$

Which is the same update for λ as that obtained using inverse iteration, see (1.29). What remains now is to give the implicit determinant method's analogue for the eigenvector update which we explain below.

Set $(\mathbf{A}-\lambda\mathbf{I})\mathbf{z}=\mathbf{b}$ in (1.32) and expand the first row to obtain

$$(\mathbf{A}-\lambda\mathbf{I})\mathbf{x}(\lambda)+f(\lambda)(\mathbf{A}-\lambda\mathbf{I})\mathbf{z}=\mathbf{0}, \quad \text{then} \quad \mathbf{x}(\lambda)=-f(\lambda)\mathbf{z}.$$

Observe that because the second row of (1.32) implies $\mathbf{c}^T\mathbf{x}(\lambda)=1$, then by premultiplying both sides of $\mathbf{x}(\lambda)=f(\lambda)\mathbf{z}$ by $\mathbf{c}^T$ simplifies to

$$f(\lambda)=-\frac{1}{\mathbf{c}^T\mathbf{z}}, \quad \text{and} \quad \mathbf{x}(\lambda)=\frac{\mathbf{z}}{\mathbf{c}^T\mathbf{z}}.$$

We conclude this section by saying that the implicit determinant method is an inefficient way of carrying out inverse iteration. This is because it involves two linear system solves at each iteration. However, one advantage of the implicit determinant method over inverse iteration is that it converges quadratically when the dimension of the nullspace of $(\mathbf{A} - \lambda^*\mathbf{I})$ is one, which includes the case when λ^* is a defective eigenvalue. Inverse iteration converges with $|\lambda^{(k)} - \lambda^*| = \mathcal{O}(1/k)$ as $k \to \infty$, $k \in \mathbb{N}$ when λ^* is a defective eigenvalue as illustrated in [63] (see, also [11]). Other advantages of the implicit determinant method will be seen in Chapters 2 and 3.

In the next section, we present some background theory on the Gauss-Newton method, which is used several times in the chapters ahead.

1.5 Background: The Gauss-Newton Method

In this section, we present background materials on the Gauss-Newton method for solving over- and under-determined systems of nonlinear equations. The motivation for these discussions comes from the fact that in Section 2.5 of Chapter 2, we will solve an over-determined system of 4-real nonlinear equations in 3 real unknowns, which arises from the theory on the coalescence of two complex eigenvalues to form a 2-dimensional Jordan block in a parameter-dependent matrix. Also, in Chapter 3, we will solve a system of $(2n + 3)$ real nonlinear equations in $(2n + 2)$ real unknowns arising from the theory of the nearest defective matrix problem. We present the theory of under-determined system of nonlinear equations because it will be applied in Chapter 4 as a key theoretical tool for solving the generalised eigenvalue problem.

1.5.1 Over-Determined Systems of Nonlinear Equations

This subsection considers the solution of over-determined nonlinear system of equations in which the number of equations is more than the number of unknowns.

Let $\mathbf{F} : \mathbb{R}^n \mapsto \mathbb{R}^m$ for $m > n$. Consider the problem of finding a solution to the over-determined nonlinear system of equations $\mathbf{F}(\mathbf{w}) = \mathbf{0}$. Define g :

$\mathbb{R}^n \mapsto \mathbb{R}$ [46, p. 267] as

$$g(\mathbf{w}) = \frac{1}{2}\mathbf{F}(\mathbf{w})^T\mathbf{F}(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{m} f_i(\mathbf{w})^2. \tag{1.41}$$

A minimizer of (1.41) for $\mathbf{w} \in \mathbb{R}^n$ is the least squares solution of the over-determined system of nonlinear equations $\mathbf{F}(\mathbf{w}) = \mathbf{0}$ [46]. Thus,

$$\begin{aligned}
\nabla g(\mathbf{w}) &= \sum_{i=1}^{m} f_i(\mathbf{w})\nabla f_i(\mathbf{w}) \\
&= \begin{bmatrix} \nabla f_1(\mathbf{w}) & \nabla f_2(\mathbf{w}) & \cdots & \nabla f_m(\mathbf{w}) \end{bmatrix} \begin{bmatrix} f_1(\mathbf{w}) \\ f_2(\mathbf{w}) \\ \vdots \\ f_m(\mathbf{w}) \end{bmatrix} \\
&= \begin{bmatrix} \frac{\partial f_1(\mathbf{w})}{\partial w_1} & \frac{\partial f_2(\mathbf{w})}{\partial w_1} & \cdots & \frac{\partial f_m(\mathbf{w})}{\partial w_1} \\ \frac{\partial f_1(\mathbf{w})}{\partial w_2} & \frac{\partial f_2(\mathbf{w})}{\partial w_2} & \cdots & \frac{\partial f_m(\mathbf{w})}{\partial w_2} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_1(\mathbf{w})}{\partial w_n} & \frac{\partial f_2(\mathbf{w})}{\partial w_n} & \cdots & \frac{\partial f_m(\mathbf{w})}{\partial w_n} \end{bmatrix} \begin{bmatrix} f_1(\mathbf{w}) \\ f_2(\mathbf{w}) \\ \vdots \\ f_m(\mathbf{w}) \end{bmatrix} \\
&= [\mathbf{F}_{\mathbf{w}}(\mathbf{w})]^T\mathbf{F}(\mathbf{w}),
\end{aligned} \tag{1.42}$$

where the Jacobian $\mathbf{F}_{\mathbf{w}}(\mathbf{w}) \in \mathbb{R}^{m\times n}$ is assumed to be of full rank. Hence, finding a stationary point of (1.41) is equivalent to finding the zeros of

$$\nabla g(\mathbf{w}) = [\mathbf{F}_{\mathbf{w}}(\mathbf{w})]^T\mathbf{F}(\mathbf{w}) = \mathbf{0}. \tag{1.43}$$

Differentiating (1.42) again we obtain

$$\begin{aligned}
\nabla^2 g(\mathbf{w}) &= \sum_{i=1}^{m} \Big(\nabla f_i(\mathbf{w}) \nabla f_i(\mathbf{w})^T + f_i(\mathbf{w}) \nabla^2 f_i(\mathbf{w}) \Big) \\
&= [\mathbf{F}_{\mathbf{w}}(\mathbf{w})]^T \mathbf{F}_{\mathbf{w}}(\mathbf{w}) + R(\mathbf{w}),
\end{aligned}$$

where

$$R(\mathbf{w}) = \sum_{i=1}^{m} f_i(\mathbf{w})\nabla^2 f_i(\mathbf{w}) = \sum_{i=1}^{m} f_i(\mathbf{w}) \begin{bmatrix} \frac{\partial^2 f_i(\mathbf{w})}{\partial w_1^2} & \frac{\partial^2 f_i(\mathbf{w})}{\partial w_1 \partial w_2} & \cdots & \frac{\partial^2 f_i(\mathbf{w})}{\partial w_1 \partial w_n} \\ \frac{\partial^2 f_i(\mathbf{w})}{\partial w_2 \partial w_1} & \frac{\partial^2 f_i(\mathbf{w})}{\partial w_2^2} & \cdots & \frac{\partial f_i(\mathbf{w})}{\partial w_2 \partial w_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial^2 f_i(\mathbf{w})}{\partial w_n \partial w_1} & \frac{\partial^2 f_i(\mathbf{w})}{\partial w_n \partial w_2} & \cdots & \frac{\partial^2 f_i(\mathbf{w})}{\partial w_n^2} \end{bmatrix}.$$

Newton's method applied to finding the zeros of (1.43) is

$$\begin{aligned} \mathbf{w}^{(k+1)} &= \mathbf{w}^{(k)} - [\nabla^2 g(\mathbf{w}^{(k)})]^{-1} \nabla g(\mathbf{w}^{(k)}) \\ &= \mathbf{w}^{(k)} - \big([\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^T \mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)}) + R(\mathbf{w}^{(k)})\big)^{-1} [\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^T \mathbf{F}(\mathbf{w}^{(k)}). \end{aligned} \tag{1.44}$$

The second term on the right hand side of (1.42) implies solving for $\Delta\mathbf{w}^{(k)}$ in

$$\big([\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^T \mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)}) + R(\mathbf{w}^{(k)})\big)\Delta\mathbf{w}^{(k)} = -[\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^T \mathbf{F}(\mathbf{w}^{(k)}),$$

and adding it to $\mathbf{w}^{(k)}$ to obtain $\mathbf{w}^{(k+1)}$. If we exclude[2] the second-order term of $\nabla^2 g(\mathbf{w})$ [42] in (1.44), then we have the Gauss-Newton method,

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} - \big([\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^T \mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\big)^{-1} [\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^T \mathbf{F}(\mathbf{w}^{(k)}). \tag{1.45}$$

In practice, we solve for $\Delta\mathbf{w}^{(k)}$ in

$$[\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^T \mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\Delta\mathbf{w}^{(k)} = -[\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^T \mathbf{F}(\mathbf{w}^{(k)}), \tag{1.46}$$

and update

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \Delta\mathbf{w}^{(k)}. \tag{1.47}$$

We remark that, (1.46) reminds us of the least squares method for minimizing $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|$ using the normal equations and can be solved using several methods (see, for example, Trefethen [60, pp. 77-84]). We will concentrate on using the

[2]When $R(\mathbf{w}^*) = \mathbf{0}$, [16, p. 222], this occurs when we have a zero-residual problem, that is, if $\mathbf{F}(\mathbf{w}^*) = \mathbf{0}$, then the sequence of iterates generated by the Gauss-Newton method converges quadratically. If $R(\mathbf{w}^*)$ is small in comparism to $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^*)^T \mathbf{F}_{\mathbf{w}}(\mathbf{w}^*)$, then the Gauss-Newton iterates converges linearly. If $R(\mathbf{w}^*)$ is too large, then the Gauss-Newton iterates may not converge at all.

QR factorization in finding a solution to (1.46). This entails finding the reduced QR factorization of $\mathbf{F}_{\mathbf{w}}(\mathbf{w}) = \mathbf{QR}$, where $\mathbf{Q} \in \mathbb{R}^{m\times n}$ is unitary and $\mathbf{R} \in \mathbb{R}^{n\times n}$ is upper triangular, and substitute into (1.46). Consequently,

$$\mathbf{R}^T\mathbf{Q}^T\mathbf{QR}\Delta\mathbf{w}^{(k)} = -\mathbf{R}^T\mathbf{Q}^T\mathbf{F}(\mathbf{w}^{(k)}), \text{ and } \mathbf{R}^T\mathbf{R}\Delta\mathbf{w}^{(k)} = -\mathbf{R}^T\mathbf{Q}^T\mathbf{F}(\mathbf{w}^{(k)}).$$

Thus, if $\mathbf{R}$ is nonsingular, then by multiplying both sides by the inverse of $\mathbf{R}$ transposed, yields

$$\mathbf{R}\Delta\mathbf{w}^{(k)} = -\mathbf{Q}^T\mathbf{F}(\mathbf{w}^{(k)}). \tag{1.48}$$

So we solve a triangular system of n equations for the n unknowns $\Delta\mathbf{w}^{(k)}$. Upon solving the linear system of equations for $\Delta\mathbf{w}^{(k)}$, we substitute $\Delta\mathbf{w}^{(k)}$ into (1.47) to obtain $\mathbf{w}^{(k+1)}$.

The following theoretical discussion on the solution of under-determined system of nonlinear equations will be used later in Chapter 4.

1.5.2 Under-Determined Systems of Nonlinear Equations

Let $\mathbf{F} : \mathbb{R}^n \to \mathbb{R}^m$ where $m < n$. Consider the problem of solving the following under-determined system of nonlinear equations $\mathbf{F}(\mathbf{w}) = \mathbf{0}$. In order to solve the system of nonlinear equations, we first linearize it. By a linearization technique (see, [46, pp.181-185]), it is not difficult to see that for $k = 0, 1, 2, \ldots$

$$\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\Delta\mathbf{w}^{(k)} = -\mathbf{F}(\mathbf{w}^{(k)}), \tag{1.49}$$

which is a sequence of under-determined linear system of equations where the Jacobian $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$ is assumed to be of full rank. The least squares solution $\Delta\mathbf{w}^{(k)}$ of minimum norm to the under-determined linear system of equations (1.49) is

$$\Delta\mathbf{w}^{(k)} = -\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^{\dagger}\mathbf{F}(\mathbf{w}^{(k)}); \tag{1.50}$$

where

$$\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^{\dagger} = \mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^T[\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^T]^{-1},$$

is the Moore-Penrose pseudo-inverse of $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$ (see also, [40, p. 143]). Therefore, we obtain

$$\Delta\mathbf{w}^{(k)} = -\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^{\dagger}\mathbf{F}(\mathbf{w}^{(k)}); \quad \text{and} \quad \mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \Delta\mathbf{w}^{(k)}, \tag{1.51}$$

which is the local Gauss-Newton method [17, pp. 221-222]. In actual computation, in solving $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\Delta\mathbf{w}^{(k)} = -\mathbf{F}(\mathbf{w}^{(k)})$, we do not compute the Moore-Penrose pseudo-inverse of $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$ explicitly. Rather, we find the reduced QR factorization $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^{T} = \mathbf{QR}$ where $\mathbf{Q}$ is a real m by n matrix and $\mathbf{R}$ is an n by n real matrix, which means that $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)}) = \mathbf{R}^{T}\mathbf{Q}^{T}$. Thus, (1.49) becomes $\mathbf{R}^{T}\mathbf{Q}^{T}\Delta\mathbf{w}^{(k)} = -\mathbf{F}(\mathbf{w}^{(k)})$. By letting $\mathbf{g}^{(k)} = \mathbf{Q}^{T}\Delta\mathbf{w}^{(k)}$ we obtain, $\mathbf{R}^{T}\mathbf{g}^{(k)} = -\mathbf{F}(\mathbf{w}^{(k)})$. Computationally, we first solve the upper-triangular system

$$\mathbf{R}^{T}\mathbf{g}^{(k)} = -\mathbf{F}(\mathbf{w}^{(k)}),$$

for the unknown vector $\mathbf{g}^{(k)}$. With the computed value of $\mathbf{g}^{(k)}$, we then solve

$$\mathbf{Q}^{T}\Delta\mathbf{w}^{(k)} = \mathbf{g}^{(k)},$$

for $\Delta\mathbf{w}^{(k)}$. But after premultiplying both sides of the above equation by $\mathbf{Q}$, we obtain the solution to (1.49) as

$$\Delta\mathbf{w}^{(k)} = \mathbf{Q}\mathbf{g}^{(k)}.$$

Since $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$ is assumed to be of full rank, $\mathbf{R}$ is invertible. Hence, the solution to (1.49) can also be expressed as $\Delta\mathbf{w}^{(k)} = -\mathbf{QR}^{-T}\mathbf{F}(\mathbf{w}^{(k)})$. The sequence of iterates $\{\mathbf{w}^{(k)}\}$ generated by the local Gauss-Newton method converges quadratically if $\mathbf{F}(\mathbf{w}^{*}) = \mathbf{0}$ (see for example, [49, p. 44], [46, p. 409], [34, p. 57], [18, p. 9], [46, pp. 412-413]). One 'fundamental' property of the minimum norm solution (1.50) which we will use in Chapter 4 is given in the following lemma.

Lemma 1.5.1. *[10, p. 6] Let* $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$ *be of full rank m. If*

$$\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\Delta\mathbf{w}^{(k)} = -\mathbf{F}(\mathbf{w}^{(k)}),$$

is an under-determined linear system of equations, then its least squares solution

$$\Delta \mathbf{w}^{(k)} = -\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^T[\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^T]^{-1}\mathbf{F}(\mathbf{w}^{(k)}),$$

is orthogonal to the nullspace of $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$.

Proof: By definition, if $\boldsymbol{n}^{(k)}$ is in the nullspace of $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$, then

$$\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\boldsymbol{n}^{(k)} = \mathbf{0}.$$

Thus,

$$\begin{aligned} {\boldsymbol{n}^{(k)}}^T \Delta \mathbf{w}^{(k)} &= -{\boldsymbol{n}^{(k)}}^T \mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^T[\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^T]^{-1}\mathbf{F}(\mathbf{w}^{(k)}) \\ &= -[\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\boldsymbol{n}^{(k)}]^T[\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})^T]^{-1}\mathbf{F}(\mathbf{w}^{(k)}), \end{aligned}$$

and because $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\boldsymbol{n}^{(k)} = \mathbf{0}$, ${\boldsymbol{n}^{(k)}}^T \Delta \mathbf{w}^{(k)} = 0$. This shows that $\Delta \mathbf{w}^{(k)}$ is orthogonal to the nullspace of $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$. ∎

In the next section, we give a literature survey of Newton's method and inverse iteration with a complex shift.

1.6 Survey of Newton's Method and Inverse Iteration with Complex Shift

This section surveys the contributions of Ruhe [51] and Tisseur [59] to the solution of nonlinear eigenvalue problems using Newton's method and inverse iteration as well as Parlett and Saad's [48]. The main point is that both [51] and [59] use two different differentiable normalisations: (1.55) and (1.58), while in Chapter 4 we analyse the natural extension of the distance norm, which is a non differentiable normalisation and so leads to interesting theoretical questions.

Let $\mathbf{T}(\lambda)$ be a parameter-dependent n by n matrix whose entries are analytic functions of the complex number λ [51]. In this section, we give a brief survey on previous approaches used to compute the eigenpair (ϕ, λ) from the

eigenvalue problem

$$\mathbf{T}(\lambda)\boldsymbol{\phi} = \mathbf{0}, \tag{1.52}$$

where $\boldsymbol{\phi} \in \mathbb{C}^n \backslash \{\mathbf{0}\}$. The standard eigenvalue problem $\mathbf{A}\boldsymbol{\phi} = \lambda\boldsymbol{\phi}$, is a special case of (1.52) if [51, p. 674]

$$\mathbf{T}(\lambda) = \mathbf{A} - \lambda\mathbf{I}, \tag{1.53}$$

or the generalised eigenvalue problem if

$$\mathbf{T}(\lambda) = \mathbf{A} - \lambda\mathbf{B}. \tag{1.54}$$

In order to apply Newton's method to (1.52), Ruhe in [51, pp. 677-678], added the normalisation

$$\mathbf{c}^H\boldsymbol{\phi} = 1, \tag{1.55}$$

where $\mathbf{c}$ is a fixed nonzero vector and obtained the following system of nonlinear $(n+1)$ equations in $(n+1)$ unknowns $\mathbf{w} = [\boldsymbol{\phi}, \lambda]^T$,

$$\mathbf{F}(\mathbf{w}) = \begin{bmatrix} \mathbf{T}(\lambda)\boldsymbol{\phi} \\ \mathbf{c}^H\boldsymbol{\phi} - 1 \end{bmatrix} = \mathbf{0}. \tag{1.56}$$

By an application of Newton's method to the nonlinear eigenvalue problem above, we have

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} - [\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})]^{-1}\mathbf{F}(\mathbf{w}^{(k)}), \quad \text{for } k = 0, 1, 2, \cdots,$$

where in this case the Jacobian

$$\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)}) = \begin{bmatrix} \mathbf{T}(\lambda^{(k)}) & \mathbf{T}'(\lambda^{(k)})\boldsymbol{\phi}^{(k)} \\ \mathbf{c}^H & 0 \end{bmatrix}.$$

In a manner analogous to the discussion following (1.45), we can write the second term on the right hand sides of $\mathbf{w}^{(k+1)}$ as $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})\Delta\mathbf{w}^{(k)} = -\mathbf{F}(\mathbf{w}^{(k)})$ or

$$\begin{bmatrix} \mathbf{T}(\lambda^{(k)}) & \mathbf{T}'(\lambda^{(k)})\boldsymbol{\phi}^{(k)} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}^{(k+1)} - \boldsymbol{\phi}^{(k)} \\ \lambda^{(k+1)} - \lambda^{(k)} \end{bmatrix} = - \begin{bmatrix} \mathbf{T}(\lambda^{(k)})\boldsymbol{\phi}^{(k)} \\ \mathbf{c}^H\boldsymbol{\phi}^{(k)} - 1 \end{bmatrix}. \tag{1.57}$$

After expanding along the first row and if $\phi^{(k)}$ is normalised as in (1.55), then one obtains [51, p. 678]

$$\begin{aligned} -\mathbf{T}(\lambda^{(k)})\phi^{(k+1)} &= (\lambda^{(k+1)} - \lambda^{(k)})\mathbf{T}'(\lambda^{(k)})\phi^{(k)} \\ \mathbf{c}^H\phi^{(k+1)} &= 1. \end{aligned}$$

It can be easily deduced that Newton's method above is equivalent to a non-linear version of inverse iteration below as [51, p. 678]

$$\begin{aligned} \mathbf{T}(\lambda^{(k)})\mathbf{v}^{(k+1)} &= \mathbf{T}'(\lambda^{(k)})\phi^{(k)} \\ \lambda^{(k+1)} &= \lambda^{(k)} - \mathbf{c}^H\phi^{(k)}/(\mathbf{c}^H\mathbf{v}^{(k+1)}) \\ \phi^{(k+1)} &= S\mathbf{v}^{(k+1)}, \end{aligned}$$

S is a normalisation constant.

Parlett and Saad in [48], studied inverse iteration with a complex shift $\sigma = \alpha + i\beta$ where α and β are real. They showed that by replacing the shifted complex system $(\mathbf{A} - \sigma\mathbf{B})\phi = \mathbf{B}\varphi$, with a real one, the size of the problem is doubled, where $\varphi = \varphi_1 + i\varphi_2$, $\phi = \phi_1 + i\phi_2$ for $\varphi_1, \varphi_2, \phi_1, \phi_2 \in \mathbb{R}^n$ and $i = \sqrt{-1}$ is the imaginary unit of a complex number. This is because solving a complex linear system of equations takes twice the storage and is roughly three times the cost of solving a real system [38]. When real arithmetic rather than complex arithmetic is used, we lose any band structure in $\mathbf{A}$ and $\mathbf{B}$ [48]. The numerical examples in [48], show linear convergence to the eigenvalue closest to the fixed shift.

Next, Tisseur in [59] considered the symmetric definite generalised eigenvalue problem $\mathbf{A}\phi = \lambda\mathbf{B}\phi$, $\lambda \in \mathbb{R}$ as a special case of (1.52) with $\mathbf{T}(\lambda)$ defined as (1.54), where $\mathbf{A}$ is symmetric and $\mathbf{B}$ is symmetric positive definite but with the real normalisation

$$\tau\mathbf{e}_s^T\phi = \tau; \quad \text{for some fixed s,} \tag{1.58}$$

where $\tau = \max(\|\mathbf{A}\|, \|\mathbf{B}\|)$, (see, for example, [59, p. 1049]) and $\mathbf{e}_j$ is the jth column of the identity matrix. The real scalar τ is introduced to scale $\mathbf{F}(\mathbf{w})$ and

$\mathbf{F_w}(\mathbf{w})$ when $\mathbf{A}$ and $\mathbf{B}$ are multiplied by a scalar. In this case,

$$\mathbf{F}(\mathbf{w}) = \begin{bmatrix} (\mathbf{A} - \lambda\mathbf{B})\phi \\ \tau\mathbf{e}_s^T\phi - \tau \end{bmatrix}, \quad \text{and} \quad \mathbf{F_w}(\mathbf{w}) = \begin{bmatrix} (\mathbf{A} - \lambda\mathbf{B}) & -\mathbf{B}\phi \\ \tau\mathbf{e}_s^T & 0 \end{bmatrix}.$$

Tisseur [59], showed that the Jacobian $\mathbf{F_w}(\mathbf{w})$ above is singular at the root if and only if λ^* is a finite multiple eigenvalue of the pencil $(\mathbf{A}, \mathbf{B})$. The main result in [59] is Theorem 2.4 [59, pp. 1044-1046]. It shows that if the linear system to be solved is not too ill conditioned, the solver is not completely unstable, the Jacobian is approximated accurately enough and we have a good initial guess very close to the solution, then the norm of the residual reduces after one step of Newton's method in floating point arithmetic. Tisseur, also examined how fixed and mixed precision iterative refinement affect the computed residual from Newton's method and remarked "the use of extended precision for computing the residual has no effect on the rate of convergence of Newton's method."

In addition, it was shown numerically in [59, pp. 1053-1054] that if Newton's method is applied in floating point arithmetic with mixed precision iterative refinement, the linear solver is unstable and there are inaccuracies in computing the Jacobian, then this may affect the rate of convergence of Newton's method but not the accuracy and stability of the computed eigenvalues.

In the next section, we describe the structure of this thesis.

1.7 Structure of this Thesis

In this section we give a recap of the content of this thesis. Our approach for finding the values of γ^* such that two eigenvalues λ_1 and λ_2, say, of $\mathbf{A}(\gamma^*)$ coalesce at λ^*, is to extend the implicit determinant method of Spence and Poulton [55] to finding the values of γ^* and λ^* such that $\left(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\right)$ has a 2-dimensional Jordan block corresponding to a zero eigenvalue. When λ^* is real, this involves the solution of two real system of nonlinear equations for the two real unknowns γ^* and λ^*. However, when λ^* is complex, we write $\lambda^* = \alpha^* + i\beta^*$, where α^*, β^* are real. This reduces to using the Gauss-Newton method to solve a real over-determined system of four nonlinear equations for

the three real unknowns α^*, β^* and γ^*. In both the real and complex cases, our approach gives quadratic convergence and results of numerical experiments are given which confirm the theory. These results are discussed in Chapter 2.

In Chapter 3, we present two approaches for solving the nearest defective matrix problem which are more computationally efficient than those proposed by Alam & Bora [4]. The first approach for solving the question posed by Wilkinson, is to extend the implicit determinant method of Spence and Poulton [55] to find parameter values for which a certain Hermitian matrix is singular subject to a constraint. The application of the extended version of the implicit determinant method, results in using Newton's method to solve a real system of three nonlinear equations for the three real unknowns, α, β and ε, where $z = \alpha + i\beta$ and ε is the distance between the simple matrix $\mathbf{A}$ and the defective $\mathbf{B}$. This part of Chapter 3 has been submitted for publication (see, [2]). The second approach for solving the nearest defective matrix problem is to use the Gauss-Newton method to solve a real system of $(2n+3)$ nonlinear equations for $(2n+2)$ real unknowns. We only describe the later method for the case in which z is real.

Finally, in Chapter 4, we consider the numerical solution of a nonsymmetric eigenvalue problem $\mathbf{A}\phi = \lambda\mathbf{B}\phi$ where $\mathbf{A}$ is nonsymmetric and $\mathbf{B}$ is symmetric positive definite and λ is complex. While Ruhe [51] and Tisseur [59] used the differentiable normalisations $\mathbf{c}^H\phi = 1$, and $\mathbf{e}_s^T\phi = 1$ respectively, we show that the generalisation of the usual 2-norm normalisation to the complex case still gives quadratic convergence even though the normalisation $\phi^H\mathbf{B}\phi = 1$ is not differentiable. This then leads to several interesting theoretical questions. We conclude by ignoring the fact that the normalisation is non differentiable and solved the resulting n complex and one real nonlinear equations for $(n+1)$ unknowns. Numerical experiments are given to back up our theoretical discussion.

Throughout this thesis, we try to be as consistent as possible in the use of notations. In all numerical computations, we used `Matlab`® Version 7.9.0.529 (R2009b).

CHAPTER 2

Implicit Determinant Method and the Computation of a 2-Dimensional Jordan Block in a Parameter Dependent Matrix

2.1 Introduction

Let $\mathbf{A}(\gamma)$ be a real n by n nonsymmetric matrix with a real parameter γ such that $\mathbf{A}(\gamma)$ is at least twice continuously differentiable with respect to γ. In this chapter, we present an extension of the implicit determinant method of Spence and Poulton [55] for the numerical computation of a 2-dimensional Jordan block in a parameter-dependent nonsymmetric matrix. A similar approach has been applied by Freitag and Spence [22] to solve a distance to instability problem for the special case where $\mathbf{A}(\gamma)$ is a Hamiltonian matrix. We will also use the implicit determinant method in the next chapter to derive a method for computing a nearby defective matrix.

Let $\lambda^* \in \mathbb{C}$ be an eigenvalue of $\mathbf{A}(\gamma^*)$. In this chapter, we consider the problem of finding γ^* such that $\mathbf{A}(\gamma^*)$ has a 2-dimensional Jordan block corresponding to the eigenvalue λ^*.

Jordan blocks are important for a number of reasons. First of all, as discussed in Chapter 1, matrices that have a Jordan block are related to sensitivity of eigendecompositions (see, for example, [14]). Secondly, they arise in applications. For example, two eigenvalues coalesce to form a 2-dimensional Jordan

block in a supersonic panel flutter (see, [53]). Thirdly, as illustrated by Dodson *et al.*, in [19], 2-dimensional Jordan blocks arise in power systems dynamics to determine when two damped oscillatory modes coalesce as power system parameters *e.g.*, power transfer and generator redispatch, change. This means that the linearized power system has two complex conjugate eigenvalues that coalesce in both damping and frequency *i.e.*, real and imaginary parts coalesce respectively.

In [22], Freitag and Spence extended the method of Spence and Poulton to a special class of parameter-dependent Hamiltonian matrices by computing a 2-dimensional Jordan block to solve a distance to instability problem. This chapter considers a more general setting than in [22] and also extends to the large, sparse nonsymmetric matrix case. The main mathematical tools used in this chapter are Keller's [33] ABCD Lemma, the Gauss-Newton method and the Block Elimination Mixed method (BEM) (see, for example, [25], [27]) for the efficient solution of bordered linear systems of equations.

This chapter is structured as follows: In Section 2.2, we describe an extended version of the implicit determinant method for the computation of a 2-dimensional Jordan block when λ^*, the eigenvalue corresponding to the Jordan block, is real, and present the main result, namely, Theorem 2.2.1. In Section 2.3, we present results of numerical experiments which support the theory. For an efficient solution of the bordered linear system of equations that arise from discretised partial differential equations, in Section 2.4, we describe the Block Elimination Mixed method ([25], [27]).

When $\mathbf{A}(\gamma)$ has a special structure, for example, tridiagonal or block tridiagonal, BEM takes advantage of this structure. Note that in solving block tridiagonal systems, it is efficient to use the block Thomas algorithm (see, for example [31, pp. 58-61]). As a result of this, in Subsection 2.4.3, we briefly describe the block Thomas algorithm for solving block tridiagonal systems. Finally, in Section 2.5, we extend the theory of the implicit determinant method of Spence and Poulton [55] to compute a 2-dimensional Jordan block in a parameter-dependent nonsymmetric matrix when λ^* is complex. Throughout this chapter, we assume that γ is real.

2.2 The Implicit Determinant Method for a Real 2-Dimensional Jordan Block in a Parameter-Dependent Matrix

In this section, we describe an application of the implicit determinant method of Spence and Poulton [55, p. 71], to a parameter-dependent nonsymmetric matrix $\mathbf{A}(\gamma)$. We first assume that λ^* is real, so to be precise, the problem is to find real λ^* and γ^* such that $\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}$ has a 2-dimensional Jordan block corresponding to a zero eigenvalue. This is presented in the main result of this section, namely, Theorem 2.2.1. We present a Newton-based algorithm for computing the pair (λ^*, γ^*) in Subsection 2.2.1. In Subsection 2.2.2, we describe the eigenvalue behaviour of $\mathbf{A}(\gamma)$ for γ near γ^* by applying standard bifurcation theory. This is then followed by a brief description on how to choose optimal starting values as well as stopping criteria for the algorithm. A condition for the algorithm to converge is given in Theorem 2.2.2.

Let $\mathbf{A}(\gamma)$ be a real parameter-dependent matrix where $\mathbf{A}(\gamma)$ is at least twice continuously differentiable. In order to find the values of λ^* and γ^* such that $\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}$ has a 2-dimensional Jordan block corresponding to the zero eigenvalue, we consider the following problem (see also [55, (15), (26)]),

$$\begin{bmatrix} \mathbf{A}(\gamma) - \lambda\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}, \tag{2.1}$$

where $\mathbf{b}$ and $\mathbf{c}$ are real $n-$component constant vectors, $f \in \mathbb{R}$ and $\mathbf{x} \in \mathbb{R}^n \backslash \{\mathbf{0}\}$.

The next result is an application of Keller's [33] ABCD Lemma to (2.1) and it shows that the matrix in (2.1) is nonsingular at the root under certain conditions.

Lemma 2.2.1. *Let* $(\mathbf{x}^*, \lambda^*, \gamma^*)$ *solve* (2.1). *Let zero be an eigenvalue of* $\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}$ *corresponding to a 2-dimensional Jordan block and* $\operatorname{rank}(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}) = n - 1$, *then*

$$\mathbf{M}(\lambda^*, \gamma^*) = \begin{bmatrix} \mathbf{A}(\gamma^*) - \lambda^*\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix}, \tag{2.2}$$

is nonsingular if and only if ${\psi^*}^T\mathbf{b} \neq 0$ *for all* $\psi^* \in \mathcal{N}\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)^T \backslash \{\mathbf{0}\}$ *and*

$\mathbf{c}^T\boldsymbol{\phi}^* \neq 0$ *for all* $\boldsymbol{\phi}^* \in \mathcal{N}\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\backslash\{\mathbf{0}\}$.

Proof: Let $\boldsymbol{\psi}^{*T}\mathbf{b} \neq 0$ for all $\boldsymbol{\psi}^* \in \mathcal{N}\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)^T\backslash\{\mathbf{0}\}$ and $\mathbf{c}^T\boldsymbol{\phi}^* \neq 0$ for all $\boldsymbol{\phi}^* \in \mathcal{N}\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)^T\backslash\{\mathbf{0}\}$. If we can show that $\mathbf{p}$ and q are each zero in

$$\begin{bmatrix} \mathbf{A}(\gamma^*) - \lambda^*\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix}\begin{bmatrix} \mathbf{p} \\ q \end{bmatrix} = \mathbf{0},$$

then $\mathbf{M}(\lambda^*, \gamma^*)$ is nonsingular. By expanding along the first row, we obtain

$$\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\mathbf{p} + q\mathbf{b} = \mathbf{0}.$$

Using $\big(\mathbf{A}(\gamma^*)^T - \lambda^*\mathbf{I}\big)\boldsymbol{\psi}^* = \mathbf{0}$, it then implies

$$\boldsymbol{\psi}^{*T}\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\mathbf{p} + q\boldsymbol{\psi}^{*T}\mathbf{b} = 0,$$

reduces to $q\boldsymbol{\psi}^{*T}\mathbf{b} = 0$. Therefore, $q = 0$, $\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\mathbf{p} = \mathbf{0}$ and $\mathbf{p} = \tau\boldsymbol{\phi}^*$, where τ is a real scalar. Now, $\mathbf{c}^T\mathbf{p} = \tau\mathbf{c}^T\boldsymbol{\phi}^* = 0$ is obvious by expanding along the second row. Consequently, $\tau = 0$ and $\mathbf{p} = \mathbf{0}$.

Conversely, let $\mathbf{M}(\lambda^*, \gamma^*)$ be nonsingular. Assume $\big(\mathbf{A}(\gamma^*)^T - \lambda^*\mathbf{I}\big)$ is singular and $\mathbf{c}^T\boldsymbol{\phi}^* = 0$, we want to show by contradiction that $\mathbf{c}^T\boldsymbol{\phi}^* \neq 0$. We multiply $\mathbf{M}(\lambda^*, \gamma^*)$ from the right by the nonzero vector $[\boldsymbol{\phi}^*, 0]^T$

$$\begin{bmatrix} \mathbf{A}(\gamma^*)^T - \lambda^*\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix}\begin{bmatrix} \boldsymbol{\phi}^* \\ 0 \end{bmatrix} = \begin{bmatrix} \big(\mathbf{A}(\gamma^*)^T - \lambda^*\mathbf{I}\big)\boldsymbol{\phi}^* \\ \mathbf{c}^T\boldsymbol{\phi}^* \end{bmatrix} = \mathbf{0}. \tag{2.3}$$

This shows that we have multiplied the nonsingular matrix $\mathbf{M}(\lambda^*, \gamma^*)$ by a nonzero vector to obtain the zero vector, this implies that $\mathbf{M}(\lambda^*, \gamma^*)$ is singular, a contradiction, hence $\mathbf{c}^T\boldsymbol{\phi}^* \neq 0$. Similarly, let $\boldsymbol{\psi}^{*T}\mathbf{b} = 0$, multiply $\mathbf{M}(\lambda^*, \gamma^*)$ from the left by the nonzero vector $[\boldsymbol{\psi}^*, 0]^T$ to obtain

$$\begin{bmatrix} \boldsymbol{\psi}^* & 0 \end{bmatrix}^T\begin{bmatrix} \mathbf{A}(\gamma^*)^T - \lambda^*\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\psi}^{*T}\big(\mathbf{A}(\gamma^*)^T - \lambda^*\mathbf{I}\big) & \boldsymbol{\psi}^{*T}\mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{0}^T & 0 \end{bmatrix}.$$

This shows that $\mathbf{M}(\lambda^*, \gamma^*)$ is singular, contradicting the nonsingularity of $\mathbf{M}(\lambda^*, \gamma^*)$, therefore, $\boldsymbol{\psi}^{*T}\mathbf{b} \neq 0$. ∎

It should be remarked that since $\mathbf{M}(\lambda^*, \gamma^*)$ is nonsingular, then $\mathbf{M}(\lambda, \gamma)$ is

nonsingular for λ and γ near λ^* and γ^* because $\mathbf{A}(\gamma) - \lambda\mathbf{I}$ is a smooth function of λ and γ (see also [55, p. 71]). Next, we extend Spence and Poulton's implicit determinant method [55, p. 71] for computing a 2-dimensional Jordan block in a parameter-dependent nonsymmetric matrix. It shows that there is a relationship between the zeros of $f(\lambda, \gamma)$ and the determinant of $\mathbf{A}(\gamma) - \lambda\mathbf{I}$.

Lemma 2.2.2. *Let the conditions of Lemma 2.2.1 hold, and consider the linear system* (2.1)*. Then*

1. $f = f(\lambda, \gamma)$ *and* $\mathbf{x} = \mathbf{x}(\lambda, \gamma)$,
2. $f(\lambda^*, \gamma^*) = 0$ *if and only if* $\det[\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}] = 0$,
3. *For* $\lambda = \lambda^*,\ \gamma = \gamma^*$, $\mathbf{x}(\lambda, \gamma) = \mathbf{x}(\lambda^*, \gamma^*) \in \mathcal{N}\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\backslash\{\mathbf{0}\}$.

Proof: Lemma 2.2.1 shows that $\mathbf{M}(\lambda^*, \gamma^*)$ is nonsingular. Since $\mathbf{A}(\gamma) - \lambda\mathbf{I}$ is a smooth function of λ and γ, then using the implicit function theorem (see for example, Spence and Graham [56, p. 186]), $\mathbf{M}(\lambda, \gamma)$ is nonsingular for λ and γ near λ^* and γ^*. Hence, $\mathbf{x}$ and f are smooth functions of λ and γ, so that we can write $f = f(\lambda, \gamma)$ and $\mathbf{x} = \mathbf{x}(\lambda, \gamma)$. Moreover, we can rewrite (2.1) as

$$\begin{bmatrix} \mathbf{A}(\gamma) - \lambda\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(\lambda, \gamma) \\ f(\lambda, \gamma) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \tag{2.4}$$

Thus, the first part of the theorem is proved. Using Cramer's rule (see, for example, [32, p. 414]) yields

$$f(\lambda, \gamma) = \frac{\det[\mathbf{A}(\gamma) - \lambda\mathbf{I}]}{\det \mathbf{M}(\lambda, \gamma)}, \tag{2.5}$$

(see also, (1.22)). Therefore, $f(\lambda, \gamma) = 0$ if and only if $(\mathbf{A}(\gamma) - \lambda\mathbf{I})$ is singular- which is attainable at the root. To obtain $\mathbf{x}(\lambda^*, \gamma^*)$ we substitute $f(\lambda^*, \gamma^*) = 0$ into the first equation in (2.4), giving $(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I})\mathbf{x}(\lambda^*, \gamma^*) = \mathbf{0}$. This implies $\mathbf{x}(\lambda^*, \gamma^*) = \tau\boldsymbol{\phi}^*$, where τ is real but nonzero, because $(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I})$ is singular. Hence, $\mathbf{x}(\lambda^*, \gamma^*) \in \mathcal{N}(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I})$. ∎

The above result means that we compute the zeros of $f(\lambda, \gamma)$ as a way of finding the zeros of the determinant of $\mathbf{A}(\gamma) - \lambda\mathbf{I}$. The next fundamental result

shows the condition satisfied by $f(\lambda,\gamma)$, and its partial derivatives: $f_\lambda(\lambda,\gamma)$ and $f_{\lambda\lambda}(\lambda,\gamma)$ at the root. This main result will be used in the following analysis to show that a certain 2 by 2 Jacobian matrix is nonsingular at the root.

Theorem 2.2.1. *Let $\mathbf{A}(\gamma^*)$ be a real n by n matrix with a 2-dimensional Jordan block corresponding to the real eigenvalue λ^*. If $\mathbf{b}$, $\mathbf{c}$ are chosen such that $\mathbf{M}(\lambda^*,\gamma^*)$ is nonsingular, then*

1. $f(\lambda^*,\gamma^*)=0$,
2. $f_\lambda(\lambda^*,\gamma^*)=0$,
3. $f_{\lambda\lambda}(\lambda^*,\gamma^*)\neq 0$.

Proof: The first part of the theorem follows from the second part of Lemma 2.2.2. After differentiating both sides of (2.4) with respect to λ, we obtain

$$\begin{bmatrix} \mathbf{A}(\gamma)-\lambda\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\lambda(\lambda,\gamma) \\ f_\lambda(\lambda,\gamma) \end{bmatrix} = \begin{bmatrix} \mathbf{x}(\lambda,\gamma) \\ 0 \end{bmatrix}, \tag{2.6}$$

where $\mathbf{x}_\lambda(\lambda,\gamma)=\frac{\mathrm{d}}{\mathrm{d}\lambda}\mathbf{x}(\lambda,\gamma)$ and $f_\lambda(\lambda,\gamma)=\frac{\mathrm{d}}{\mathrm{d}\lambda}f(\lambda,\gamma)$. Now, if we expand along the first n rows, then

$$\big(\mathbf{A}(\gamma)-\lambda\mathbf{I}\big)\mathbf{x}_\lambda(\lambda,\gamma)+f_\lambda(\lambda,\gamma)\mathbf{b}=\mathbf{x}(\lambda,\gamma). \tag{2.7}$$

Evaluate the above at $\lambda=\lambda^*$ and $\gamma=\gamma^*$, premultiply both sides by ${\boldsymbol{\psi}^*}^T$, for all $\boldsymbol{\psi}^*\in\mathcal{N}\big(\mathbf{A}(\gamma^*)-\lambda^*\mathbf{I}\big)^T\setminus\{\mathbf{0}\}$, to obtain

$${\boldsymbol{\psi}^*}^T\big(\mathbf{A}(\gamma^*)-\lambda^*\mathbf{I}\big)\mathbf{x}_\lambda(\lambda^*,\gamma^*)+f_\lambda(\lambda^*,\gamma^*){\boldsymbol{\psi}^*}^T\mathbf{b}={\boldsymbol{\psi}^*}^T\mathbf{x}(\lambda^*,\gamma^*),$$

which simplifies to

$$f_\lambda(\lambda^*,\gamma^*)=\frac{{\boldsymbol{\psi}^*}^T\mathbf{x}(\lambda^*,\gamma^*)}{{\boldsymbol{\psi}^*}^T\mathbf{b}}=\frac{\tau{\boldsymbol{\psi}^*}^T\boldsymbol{\phi}^*}{{\boldsymbol{\psi}^*}^T\mathbf{b}},\quad \tau\neq 0, \tag{2.8}$$

where we have used the fact from Lemma 2.2.1 that ${\boldsymbol{\psi}^*}^T\mathbf{b}\neq 0$ and $\mathbf{x}(\lambda^*,\gamma^*)=\tau\boldsymbol{\phi}^*$ by virtue of Lemma 2.2.2. But ${\boldsymbol{\psi}^*}^T\boldsymbol{\phi}^*=0$, since $\mathbf{A}(\gamma^*)$ has a 2-dimensional Jordan block. Hence, $f_\lambda(\lambda^*,\gamma^*)=0$. Thus, proving the second part of the

theorem. After substituting $f_\lambda(\lambda^*, \gamma^*) = 0$ into (2.7),

$$\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\mathbf{x}_\lambda(\lambda^*, \gamma^*) = \mathbf{x}(\lambda^*, \gamma^*).$$

This means that $\mathbf{x}_\lambda(\lambda^*, \gamma^*)$ can be taken as the generalised eigenvector $\hat{\phi}^*$ of $\mathbf{A}(\gamma^*)$, corresponding to the eigenvalue λ^* (*cf.*, equation (1.9)). Since the Jordan block has dimension 2, we have

$$\boldsymbol{\psi}^{*T}\mathbf{x}_\lambda(\lambda^*, \gamma^*) \neq 0, \tag{2.9}$$

(see also [22, (13)]). Again, by differentiating both sides of (2.6) with respect to λ,

$$\begin{bmatrix} A(\gamma) - \lambda\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\lambda\lambda}(\lambda, \gamma) \\ f_{\lambda\lambda}(\lambda, \gamma) \end{bmatrix} = \begin{bmatrix} 2\mathbf{x}_\lambda(\lambda, \gamma) \\ 0 \end{bmatrix}. \tag{2.10}$$

In a manner analogous to the analysis that led to (2.8), we obtain

$$f_{\lambda\lambda}(\lambda^*, \gamma^*) = \frac{2\boldsymbol{\psi}^{*T}\mathbf{x}_\lambda(\lambda^*, \gamma^*)}{\boldsymbol{\psi}^{*T}\mathbf{b}} \neq 0,$$

by virtue of (2.9). ∎

Conditions 1. and 2. of Theorem 2.2.1, indicate how to find the values of λ^* and γ^*, namely, set up the nonlinear system of equations

$$\mathbf{G}(\mathbf{y}) = \begin{bmatrix} f(\lambda, \gamma) \\ f_\lambda(\lambda, \gamma) \end{bmatrix} = \mathbf{0}, \tag{2.11}$$

where $\mathbf{y} = [\lambda, \gamma]^T$. Observe that because $f(\lambda, \gamma)$ and $f_\lambda(\lambda, \gamma)$ are both real, and since λ and γ are also real, this implies that (2.11) entails solving two real nonlinear equations in two real unknowns. In the next section, we present a Newton based algorithm for solving (2.11) in the form of Algorithm 3.

2.2.1 Newton based Algorithm for solving (2.11)

The aims of this subsection are: to prove that the Jacobian of (2.11) is non-singular at the root under a certain nondegeneracy condition, hence, showing quadratic convergence for close enough starting guesses, and to describe how

to implement Newton's method for $\mathbf{G}(\mathbf{y}) = \mathbf{0}$ given by (2.11). The key result in this section, Theorem 2.2.2, guarantees the quadratic convergence of the Newton based Algorithm.

We first show how to calculate the elements: $f_\gamma(\lambda,\gamma)$ and $f_{\lambda\gamma}(\lambda,\gamma)$ of the Jacobian

$$\mathbf{G}_\mathbf{y}(\mathbf{y}) = \begin{bmatrix} f_\lambda(\lambda,\gamma) & f_\gamma(\lambda,\gamma) \\ f_{\lambda\lambda}(\lambda,\gamma) & f_{\lambda\gamma}(\lambda,\gamma) \end{bmatrix}, \tag{2.12}$$

of $\mathbf{G}(\mathbf{y})$ in (2.11). The other two elements $f_\lambda(\lambda,\gamma)$ and $f_{\lambda\lambda}(\lambda,\gamma)$ can be obtained by solving (2.6) and (2.10) respectively. This will then be followed by presenting a condition under which the Jacobian (2.12) is nonsingular at the root. Algorithm 3 is given for computing λ and γ. Finally, we will describe how to choose $\mathbf{b}$ and $\mathbf{c}$ as well as state a criterion for stopping the algorithm.

After differentiating (2.4) and (2.6) with respect to γ, we obtain

$$\begin{bmatrix} \mathbf{A}(\gamma) - \lambda\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\gamma(\lambda,\gamma) \\ f_\gamma(\lambda,\gamma) \end{bmatrix} = \begin{bmatrix} -\mathbf{A}'(\gamma)\mathbf{x}(\lambda,\gamma) \\ 0 \end{bmatrix}, \tag{2.13}$$

where $\mathbf{A}'(\gamma) = \frac{\mathrm{d}}{\mathrm{d}\gamma}\mathbf{A}(\gamma)$ and

$$\begin{bmatrix} \mathbf{A}(\gamma) - \lambda\mathbf{I} & \mathbf{b} \\ \mathbf{c}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\lambda\gamma}(\lambda,\gamma) \\ f_{\lambda\gamma}(\lambda,\gamma) \end{bmatrix} = \begin{bmatrix} -\mathbf{A}'(\gamma)\mathbf{x}_\lambda(\lambda,\gamma) + \mathbf{x}_\gamma(\lambda,\gamma) \\ 0 \end{bmatrix}. \tag{2.14}$$

So for a given (λ,γ), $f_\gamma(\lambda,\gamma)$ and $f_{\lambda\gamma}(\lambda,\gamma)$ can be obtained by solving (2.13) and (2.14) respectively.

From Theorem 2.2.1, at the root (see also [22, p. 7]),

$$\mathbf{G}_\mathbf{y}(\mathbf{y}^*) = \begin{bmatrix} 0 & f_\gamma(\lambda^*,\gamma^*) \\ f_{\lambda\lambda}(\lambda^*,\gamma^*) & f_{\lambda\gamma}(\lambda^*,\gamma^*) \end{bmatrix}. \tag{2.15}$$

Accordingly,

$$\det[\mathbf{G}_\mathbf{y}(\mathbf{y}^*)] = -f_\gamma(\lambda^*,\gamma^*)f_{\lambda\lambda}(\lambda^*,\gamma^*),$$

is nonzero if $f_\gamma(\lambda^*,\gamma^*)$ is not equal to zero (using the third part of Theorem 2.2.1). In the next theorem, we use the expression for the determinant

of $\mathbf{G}_{\mathbf{y}}(\mathbf{y}^*)$ above to show that the Jacobian is nonsingular if and only if

$$\boldsymbol{\psi}^{*T}\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*,\gamma^*) \neq 0,$$

which is equivalent to $f_\gamma(\lambda^*,\gamma^*) \neq 0$.

Theorem 2.2.2. *Under the assumptions of Theorem 2.2.1, the Jacobian* $\mathbf{G}_{\mathbf{y}}(\mathbf{y}^*)$*, is nonsingular if and only if* $\boldsymbol{\psi}^{*T}\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*,\gamma^*) \neq 0$.

Proof: At the root, the first n rows of (2.13) give,

$$\big(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}\big)\mathbf{x}_\gamma(\lambda^*,\gamma^*) + f_\gamma(\lambda^*,\gamma^*)\mathbf{b} = -\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*,\gamma^*).$$

By premultiplying both sides by $\boldsymbol{\psi}^{*T}$ where $\boldsymbol{\psi}^* \in \mathcal{N}(\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I})^T \backslash \{\mathbf{0}\}$, with some simplifications, we have

$$f_\gamma(\lambda^*,\gamma^*) = -\frac{\boldsymbol{\psi}^{*T}\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*,\gamma^*)}{\boldsymbol{\psi}^{*T}\mathbf{b}}, \tag{2.16}$$

Observe that $\boldsymbol{\psi}^{*T}\mathbf{b} \neq 0$, by the assumption in Lemma 2.2.1. So, $f_\gamma(\lambda^*,\gamma^*)$ is nonzero if and only if $\boldsymbol{\psi}^{*T}\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*,\gamma^*) \neq 0$. Therefore,

$$\det[\mathbf{G}_{\mathbf{y}}(\mathbf{y}^*)] \neq 0, \quad \Longleftrightarrow \boldsymbol{\psi}^{*T}\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*,\gamma^*) \neq 0.$$

■

The above Theorem gives a condition that ensures the inverse of $\mathbf{G}_{\mathbf{y}}(\mathbf{y}^*)$ exists and Algorithm 3 is guaranteed to converge quadratically with a close enough initial guess. Next, we present Algorithm 3 which is actually Newton's method for finding the zeros of $\mathbf{G}(\mathbf{y}) = \mathbf{0}$.

Observe that because $f(\lambda,\gamma)$ and its partial derivatives can be computed using the same matrix $\mathbf{M}(\lambda,\gamma)$ defined by (2.2), but with different right hand sides, this means that only one 'LU' factorization is needed in each iteration of Algorithm 3. The stopping condition for Algorithm 3 is

$$\|\Delta\mathbf{y}^{(k)}\| \leq tol, \tag{2.18}$$

where $\Delta\mathbf{y}^{(k)} = [\Delta\lambda^{(k)}, \Delta\gamma^{(k)}]^T$ and tol is some user defined error tolerance.

Algorithm 3 Newton-based Algorithm for Computing $[\lambda^{(k)}, \gamma^{(k)}]^T$

Input: Choose $\lambda^{(0)}$, $\gamma^{(0)}$ and $\mathbf{b}, \mathbf{c} \in \mathbb{C}^n \backslash \{\mathbf{0}\}$ such that $\mathbf{M}(\lambda^{(0)}, \gamma^{(0)})$ is nonsingular, *tol*.

1: **for** $k = 0, 1, 2, \ldots$, until convergence **do**

2: Solve (2.4), to obtain $\mathbf{x}(\lambda^{(k)}, \gamma^{(k)})$ and $f(\lambda^{(k)}, \gamma^{(k)})$.

3: Use the $\mathbf{x}(\lambda^{(k)}, \gamma^{(k)})$ obtained from (2.4) in solving (2.6) for $\mathbf{x}_\lambda(\lambda^{(k)}, \gamma^{(k)})$ and $f_\lambda(\lambda^{(k)}, \gamma^{(k)})$.

4: Use the $f(\lambda^{(k)}, \gamma^{(k)})$ and $f_\lambda(\lambda^{(k)}, \gamma^{(k)})$ to form

$$\mathbf{G}(\mathbf{y}^{(k)}) = \begin{bmatrix} f(\lambda^{(k)}, \gamma^{(k)}) \\ f_\lambda(\lambda^{(k)}, \gamma^{(k)}) \end{bmatrix}.$$

5: Solve (2.10) for $\mathbf{x}_{\lambda\lambda}(\lambda^{(k)}, \gamma^{(k)})$ and $f_{\lambda\lambda}(\lambda^{(k)}, \gamma^{(k)})$ using the $\mathbf{x}_\lambda(\lambda^{(k)}, \gamma^{(k)})$ obtained from (2.6).

6: Solve (2.13) for $\mathbf{x}_\gamma(\lambda^{(k)}, \gamma^{(k)})$ and $f_\gamma(\lambda^{(k)}, \gamma^{(k)})$ using the values obtained from (2.4).

7: Using the $\mathbf{x}_\lambda(\lambda^{(k)}, \gamma^{(k)})$ and $\mathbf{x}_\gamma(\lambda^{(k)}, \gamma^{(k)})$ obtained from (2.10) and (2.13) respectively, solve (2.14) for $\mathbf{x}_{\lambda\gamma}(\lambda^{(k)}, \gamma^{(k)})$ and $f_{\lambda\gamma}(\lambda^{(k)}, \gamma^{(k)})$.

8: Form and solve the linear system of equations

$$\begin{bmatrix} f_\lambda(\lambda^{(k)}, \gamma^{(k)}) & f_\gamma(\lambda^{(k)}, \gamma^{(k)}) \\ f_{\lambda\lambda}(\lambda^{(k)}, \gamma^{(k)}) & f_{\lambda\gamma}(\lambda^{(k)}, \gamma^{(k)}) \end{bmatrix} \begin{bmatrix} \Delta\lambda^{(k)} \\ \Delta\gamma^{(k)} \end{bmatrix} = - \begin{bmatrix} f(\lambda^{(k)}, \gamma^{(k)}) \\ f_\lambda(\lambda^{(k)}, \gamma^{(k)}) \end{bmatrix}, \quad (2.17)$$

for $[\Delta\lambda^{(k)}, \Delta\gamma^{(k)}]^T$.

9: Apply Newton update

$$\begin{bmatrix} \lambda^{(k+1)} \\ \gamma^{(k+1)} \end{bmatrix} = \begin{bmatrix} \lambda^{(k)} \\ \gamma^{(k)} \end{bmatrix} + \begin{bmatrix} \Delta\lambda^{(k)} \\ \Delta\gamma^{(k)} \end{bmatrix}.$$

10: **end for**

Output: $\mathbf{y}^{(k_{\max})} = [\lambda^{(k_{\max})}, \gamma^{(k_{\max})}]^T$.

The choice of the vectors $\mathbf{b}$ and $\mathbf{c}$ should be such that ${\boldsymbol{\psi}^*}^T\mathbf{b} \neq 0$ and $\mathbf{c}^T\boldsymbol{\phi}^* \neq 0$, in agreement with the conditions of Lemma 2.2.1. We could take $\mathbf{c}$ as an estimate of $\mathbf{x}^*$, namely, the right singular vector of $\mathbf{A}(\gamma^{(0)})$ (where $\gamma^{(0)}$ is a starting guess for γ^*) corresponding to its smallest singular value and motivated by the result of Theorem 2.2.2, we could take $\mathbf{b} = \mathbf{A}'(\gamma^{(0)})\mathbf{c}$. One of the computational implications of the result of Theorem 2.2.2 is that, for $f_\gamma(\lambda^*, \gamma^*)$ to be nonzero, an optimal choice of $\mathbf{b}$ would be $\mathbf{b} = \mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*, \gamma^*)$, so that $f_\gamma(\lambda^*, \gamma^*) = -1$. However, because λ^*, γ^* and $\mathbf{x}(\lambda^*, \gamma^*)$ are not known before hand, that is why we take $\mathbf{b} = \mathbf{A}'(\gamma^{(0)})\mathbf{c}$.

The following result is an extension of the result of Spence and Poulton [54, pp. 588-589] and it shows that ${\boldsymbol{\psi}^*}^T\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*, \gamma^*) \neq 0$ if and only if

$$\frac{\mathrm{d}}{\mathrm{d}\gamma}\det[\mathbf{A}(\gamma) - \lambda\mathbf{I}]\Big|_{(\lambda,\gamma)=(\lambda^*,\gamma^*)},$$

is nonzero.

Lemma 2.2.3. *Let* $\mathbf{x}(\lambda^*, \gamma^*)$ *solve* (2.4), *such that* $\det[\mathbf{A}(\gamma^*) - \lambda^*\mathbf{I}] = 0$. *Then*

$${\boldsymbol{\psi}^*}^T\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*, \gamma^*) \neq 0, \tag{2.19}$$

if and only if

$$\frac{\mathrm{d}}{\mathrm{d}\gamma}\det[\mathbf{A}(\gamma) - \lambda\mathbf{I}]\Big|_{(\lambda,\gamma)=(\lambda^*,\gamma^*)} \neq 0. \tag{2.20}$$

Proof: Using (2.5) in the proof of Lemma 2.2.2, we obtain

$$\det[\mathbf{A}(\gamma) - \lambda\mathbf{I}] = f(\lambda, \gamma)\det\mathbf{M}(\lambda, \gamma).$$

By differentiating both sides with respect to γ and evaluating at the root, yields

$$\frac{\mathrm{d}}{\mathrm{d}\gamma}\det[\mathbf{A}(\gamma) - \lambda\mathbf{I}]\Big|_{(\lambda,\gamma)=(\lambda^*,\gamma^*)} = f_\gamma(\lambda^*, \gamma^*)\det\mathbf{M}(\lambda^*, \gamma^*).$$

It is easily seen from (2.16) that

$$\frac{\mathrm{d}}{\mathrm{d}\gamma}\det[\mathbf{A}(\gamma) - \lambda\mathbf{I}]\Big|_{(\lambda,\gamma)=(\lambda^*,\gamma^*)} = -\frac{{\boldsymbol{\psi}^*}^T\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*, \gamma^*)}{{\boldsymbol{\psi}^*}^T\mathbf{b}}\det\mathbf{M}(\lambda^*, \gamma^*),$$

if and only if

$$\boldsymbol{\psi}^{*T}\mathbf{A}'(\gamma^*)\mathbf{x}(\lambda^*,\gamma^*) \neq 0.$$

Since Lemma 2.2.1 guarantees that the determinant of $\mathbf{M}(\lambda^*,\gamma^*)$ is nonzero the result follows. ∎

Condition (2.20) shows that the determinant of $\mathbf{A}(\gamma) - \lambda\mathbf{I}$ passes through zero with a nonzero derivative at $(\lambda,\gamma) = (\lambda^*,\gamma^*)$, which is a typical nondegeneracy condition. Besides, the fact that the condition (2.19) holds is very essential for Algorithm 3 to work. This is because, it ensures that $f_\gamma(\lambda^*,\gamma^*)$ is nonzero, hence, the nonsingularity of the Jacobian (2.12) at the root.

Next, we discuss the eigenvalue structure of $\mathbf{A}(\gamma)$ near the Jordan block in the following subsection.

2.2.2 Eigenvalue Structure near the 2-Dimensional Jordan Block

In this subsection, we describe the eigenvalue behaviour of $\mathbf{A}(\gamma)$ for γ near γ^* by applying standard bifurcation theory ideas (see, for example, Example 5.1 of [56]) to (2.11). However, the techniques are fairly straightforward so we describe the analysis from first principles.

First, we write down (2.11) again for ease of reference:

$$f(\lambda,\gamma) = 0, \quad f_\lambda(\lambda,\gamma) = 0, \quad \text{for all } \lambda,$$

and we assume the following conditions are satisfied as in the previous section:

$$f(\lambda^*,\gamma^*) = 0, \tag{2.21}$$
$$f_\lambda(\lambda^*,\gamma^*) = 0, \tag{2.22}$$
$$f_\gamma(\lambda^*,\gamma^*) \neq 0, \tag{2.23}$$
$$f_{\lambda\lambda}(\lambda^*,\gamma^*) \neq 0. \tag{2.24}$$

Since $f(\lambda^*,\gamma^*) = 0$ and $f_\gamma(\lambda^*,\gamma^*) \neq 0$, the implicit function theorem [56] implies that for (λ,γ) near (λ^*,γ^*), $\gamma = \gamma(\lambda)$ and we may write

$$f\big(\lambda,\gamma(\lambda)\big) = 0.$$

Differentiation with respect to λ leads to

$$f_\lambda(\lambda,\gamma(\lambda)) + f_\gamma(\lambda,\gamma(\lambda))\frac{\mathrm{d}\gamma(\lambda)}{\mathrm{d}\lambda} = 0, \tag{2.25}$$

and evaluation at (λ^*,γ^*) shows that

$$\frac{\mathrm{d}\gamma(\lambda^*)}{\mathrm{d}\lambda} = 0,$$

using (2.22) and (2.23).

Similarly, by differentiating (2.25) with respect to λ, one obtains

$$f_{\lambda\lambda}(\lambda,\gamma(\lambda)) + \left[2f_{\lambda\gamma}(\lambda,\gamma(\lambda)) + f_{\gamma\gamma}(\lambda,\gamma(\lambda))\frac{\mathrm{d}\gamma(\lambda)}{\mathrm{d}\lambda}\right]\frac{\mathrm{d}\gamma(\lambda)}{\mathrm{d}\lambda} + f_\gamma(\lambda,\gamma(\lambda))\frac{\mathrm{d}^2\gamma}{\mathrm{d}\lambda^2}(\lambda) = 0,$$

and evaluating at $\lambda = \lambda^*$, $\gamma = \gamma^*$, yields

$$\frac{\mathrm{d}^2\gamma}{\mathrm{d}\lambda^2}(\lambda^*) = -\frac{f_{\lambda\lambda}(\lambda^*,\gamma(\lambda^*))}{f_\gamma(\lambda^*,\gamma(\lambda^*))} \neq 0,$$

using (2.23) and (2.24). Hence, local to (λ^*,γ^*) we may write the solution of $f(\lambda,\gamma(\lambda)) = 0$ as

$$\gamma(\lambda) = \gamma(\lambda^*) - \frac{1}{2}(\lambda-\lambda^*)^2\frac{f_{\lambda\lambda}(\lambda^*,\gamma(\lambda^*))}{f_\gamma(\lambda^*,\gamma(\lambda^*))} + h.o.t,$$

using Taylor series. Assuming $\dfrac{f_{\lambda\lambda}(\lambda^*,\gamma(\lambda^*))}{f_\gamma(\lambda^*,\gamma(\lambda^*))} > 0$, (it is easy to see what happens if the sign is reversed) it is possible to sketch a solution diagram of $f(\lambda,\gamma) = 0$ near (λ^*,γ^*) as follows (see Figure 2-1). Here we see that, for $\gamma < \gamma^*$, there are two real solutions of $f(\lambda,\gamma) = 0$ say, (λ_1,γ) and (λ_2,γ). This corresponds to there being two real eigenvalues say, λ_1 and λ_2 of $\mathbf{A}(\gamma)$ for $\gamma < \gamma^*$. As γ approaches γ^*, the two solutions of $f(\lambda,\gamma) = 0$ coalesce to form a unique quadratic turning point of $f(\lambda,\gamma) = 0$, which corresponds to a 2-dimensional Jordan block of $\mathbf{A}(\gamma^*)$. Since (λ^*,γ^*) is an isolated root of $f(\lambda,\gamma) = 0$, $f_\lambda(\lambda,\gamma) = 0$, $\mathbf{A}(\gamma^*)$ has an isolated 2-dimensional Jordan block at

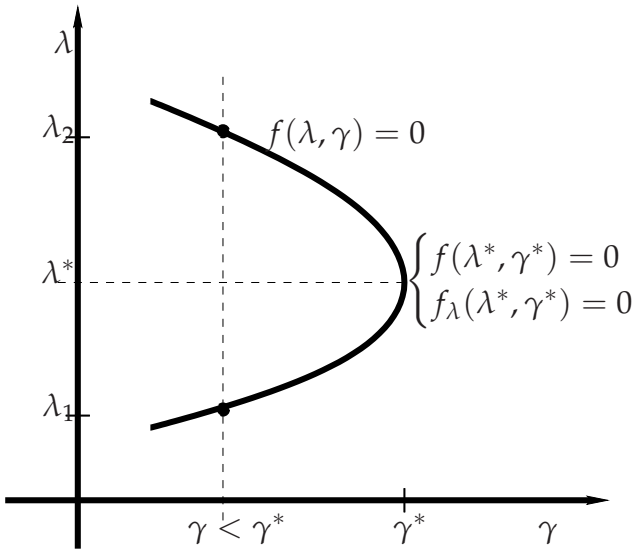

Figure 2-1: The figure above shows the path of solution of $f(\lambda,\gamma)=0$ near (λ^*,γ^*). Not drawn to scale.

$\gamma=\gamma^*$ and $\lambda=\lambda^*$. For $\gamma>\gamma^*$, there are no real solutions of $f(\lambda,\gamma)=0$. This corresponds to the fact that for $\gamma>\gamma^*$, $\mathbf{A}(\gamma)$ has two complex eigenvalues near (λ^*,γ^*). So we may sketch the eigenvalue structure of $\mathbf{A}(\gamma)$ near (λ^*,γ^*) as follows (see Figure 2-2).

Before we present the result of numerical experiments, we next discuss attainable accuracy of solving the linear systems in Algorithm 3.

2.2.3 Discussion of Attainable Accuracy

A close look at Algorithm 3 and precisely steps 8 and 9 shows that for each k, we update $\Delta\lambda^{(k)}$ and $\Delta\gamma^{(k)}$ by solving the 2 by 2 linear system of equations (2.17), where the coefficients comprising of $f_\lambda(\lambda,\gamma)$, $f_{\lambda\lambda}(\lambda,\gamma)$, $f_\gamma(\lambda,\gamma)$, $f_{\lambda\gamma}(\lambda,\gamma)$ and right hand sides $f(\lambda,\gamma)$, $f_\lambda(\lambda,\gamma)$ are subject to errors from linear solves with $\mathbf{M}(\lambda^{(k)},\gamma^{(k)})$. As a result of this, in this section, we briefly discuss the attainable accuracy of the solution computed from (2.17) and linear solves with $\mathbf{M}(\lambda^{(k)},\gamma^{(k)})$ by stating some well known perturbation results on linear systems. This will then be followed by a short theoretical discussion on iterative refinement.

The following result taken from [6, pp. 462-463] gives a bound on the at-

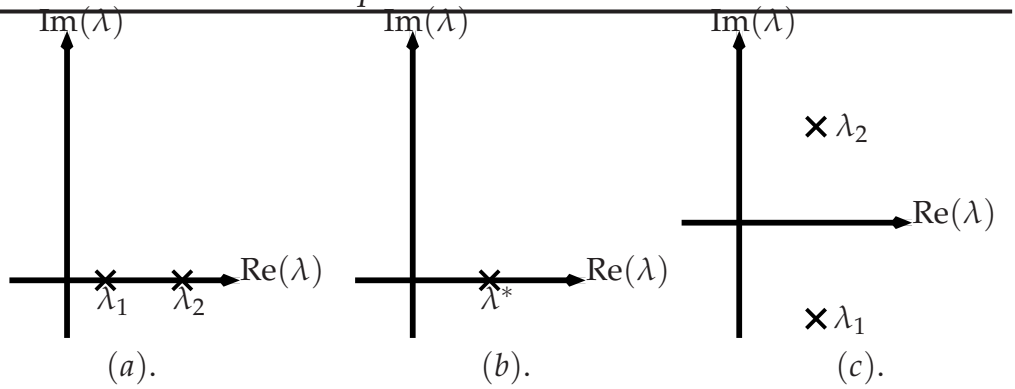

Figure 2-2: (a). For $\gamma < \gamma^*$, we see 2 real but distinct eigenvalues of $\mathbf{A}(\gamma)$. (b). When $\gamma = \gamma^*$, we see the coalescence of two distinct eigenvalues at $\lambda = \lambda^*$. (c). For $\gamma > \gamma^*$, the eigenvalues become complex conjugate eigenvalues. Not drawn to scale.

tainable accuracy of the computed solution from linear systems.

Lemma 2.2.4. *Let* $\mathbf{A}$ *be nonsingular and consider the problem of solving* $\mathbf{Ax} = \mathbf{b}$. *Let* $\Delta\mathbf{A}$ *and* $\Delta\mathbf{b}$ *be perturbations of* $\mathbf{A}$ *and* $\mathbf{b}$ *respectively. Assuming that*

$$\|\Delta\mathbf{A}\| < \frac{1}{\|\mathbf{A}^{-1}\|},$$

then $\mathbf{A} + \Delta\mathbf{A}$ *is nonsingular. Moreover, if we define* $\Delta\mathbf{x}$ *by*

$$(\mathbf{A} + \Delta\mathbf{A})(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{b} + \Delta\mathbf{b}, \tag{2.26}$$

then

$$\frac{\|\Delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \frac{\kappa(\mathbf{A})}{1 - \kappa(\mathbf{A})\|\Delta\mathbf{A}\|\|\mathbf{A}\|^{-1}} \left\{ \frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|} + \frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|} \right\}. \tag{2.27}$$

The above result [15, p. 33] shows that the relative error in the solution of $\mathbf{Ax} = \mathbf{b}$ is a multiple of the relative errors in the inputs $\mathbf{A}$ and $\mathbf{b}$, where in this case $\mathbf{A} = \mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$, $\mathbf{x}$ is $[\mathbf{x}(\lambda^{(k)}, \gamma^{(k)}), f(\lambda^{(k)}, \gamma^{(k)})]^T$ or its partial derivatives and $\mathbf{b}$ are the corresponding right hand sides.

As stated earlier in the first paragraph of this section, each of the elements in the coefficient matrix and right hand sides in the 2 by 2 system

$$\begin{bmatrix} f_\lambda(\lambda^{(k)}, \gamma^{(k)}) & f_\gamma(\lambda^{(k)}, \gamma^{(k)}) \\ f_{\lambda\lambda}(\lambda^{(k)}, \gamma^{(k)}) & f_{\lambda\gamma}(\lambda^{(k)}, \gamma^{(k)}) \end{bmatrix} \begin{bmatrix} \Delta\lambda^{(k)} \\ \Delta\gamma^{(k)} \end{bmatrix} = - \begin{bmatrix} f(\lambda^{(k)}, \gamma^{(k)}) \\ f_\lambda(\lambda^{(k)}, \gamma^{(k)}) \end{bmatrix},$$

have errors, arising from solves with $\mathbf{M}(\lambda,\gamma)$. So, in fact[1],

$$\begin{bmatrix} f_\lambda(\lambda,\gamma)+\varepsilon_{11} & f_\gamma(\lambda,\gamma)+\varepsilon_{12} \\ f_{\lambda\lambda}(\lambda,\gamma)+\varepsilon_{21} & f_{\lambda\gamma}(\lambda,\gamma)+\varepsilon_{22} \end{bmatrix} \begin{bmatrix} \Delta\lambda+\varepsilon_5 \\ \Delta\gamma+\varepsilon_6 \end{bmatrix} = - \begin{bmatrix} f(\lambda,\gamma)+\varepsilon_7 \\ f_\lambda(\lambda,\gamma)+\varepsilon_{11} \end{bmatrix}, \quad (2.28)$$

where each of the ε_{ij}'s for $i,j=1,2$ and ε_j for $j=5,6,7$ are errors in computing $f_\lambda(\lambda,\gamma)$, $f_{\lambda\lambda}(\lambda,\gamma)$, $f_\gamma(\lambda,\gamma)$, and $f_{\lambda\gamma}(\lambda,\gamma)$ *e.t.c.* From (2.28), in the light of Lemma 2.2.4, such that $\mathbf{G}_\mathbf{y}(\mathbf{y})$ is nonsingular and

$$\|\Delta\mathbf{G}_\mathbf{y}(\mathbf{y})\|\|\mathbf{G}_\mathbf{y}(\mathbf{y})^{-1}\| \le c < 1,$$

for some constant c, then

$$\left\| \begin{bmatrix} \varepsilon_5 \\ \varepsilon_6 \end{bmatrix} \right\| \Big/ \left\| \begin{bmatrix} \Delta\lambda \\ \Delta\gamma \end{bmatrix} \right\| \lesssim \kappa(\mathbf{G}_\mathbf{y}(\mathbf{y})) \Bigg(\left\| \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} \\ \varepsilon_{21} & \varepsilon_{22} \end{bmatrix} \right\| \Big/ \left\| \begin{bmatrix} f_\lambda(\lambda,\gamma) & f_\gamma(\lambda,\gamma) \\ f_{\lambda\lambda}(\lambda,\gamma) & f_{\lambda\gamma}(\lambda,\gamma) \end{bmatrix} \right\| + \left\| \begin{bmatrix} \varepsilon_7 \\ \varepsilon_{11} \end{bmatrix} \right\| \Big/ \left\| \begin{bmatrix} f(\lambda,\gamma) \\ f_\lambda(\lambda,\gamma) \end{bmatrix} \right\| \Bigg). \quad (2.29)$$

In most cases of interest here, the condition number of $\mathbf{G}_\mathbf{y}(\mathbf{y})$ is likely to be "small", so we assume that the main errors in finding $[\Delta\lambda,\Delta\gamma]^T$ arise from the errors caused by the solves with $\mathbf{M}(\lambda,\gamma)$. Hence, $[\Delta\lambda,\Delta\gamma]^T$ can only be computed to high accuracy if the ε_{ij}'s and ε_j's are small (of the order of machine precision), and the size of the condition number of $\mathbf{M}(\lambda,\gamma)$ is also small.

As will be seen in the PDE examples in the next section, a large condition number of $\mathbf{M}(\lambda,\gamma)$ impacts on the achievable accuracy of $[\Delta\lambda,\Delta\gamma]^T$. But iterative refinement seeks to overcome errors of linear systems in which the condition number of the coefficient matrix is large. This is what motivates the following brief discussion on the theory of iterative refinement.

Assuming $\mathbf{Ax}=\mathbf{b}$ has been solved by Gaussian elimination with partial pivoting and we want to improve the accuracy of the computed solution $\hat{\mathbf{x}}$. Iterative refinement (see, for example [30, p. 232]) is a method of improving the computed solution $\hat{\mathbf{x}}$ to the linear system $\mathbf{Ax}=\mathbf{b}$. This involves computing the residual of the system, $\mathbf{r}=\mathbf{b}-\mathbf{A}\hat{\mathbf{x}}$, solving $\mathbf{As}=\mathbf{r}$ and updating $\mathbf{x}=\hat{\mathbf{x}}+\mathbf{s}$.

[1]We dropped the superscripts, as in $\lambda^{(k)}$ in each of the $f(\lambda,\gamma)$ *e.t.c.*, for ease of notation.

If **r**, **s** and **x** are computed in the absence of round off errors, then **x** is the exact solution to the system. The key idea behind iterative refinement [30, p. 232] is that, if the residual **r** and **s** are computed accurately enough, then there will be some improvements in the accuracy of **x**.

Algorithm 4 is called fixed precision iterative refinement (see, for example Golub and van Loan [23]). The term "fixed precision" is so used because, both the residual **r** and the **s** are computed using the same precision. Let us recall

Algorithm 4 Fixed Precision Iterative Refinement

Input: $\mathbf{A}, \hat{\mathbf{x}}, \mathbf{b}$

1: Compute LU factorization of $\mathbf{A}$.
2: **for** $k = 1, 2, \ldots,$ **do**
3: Compute the residual $\mathbf{r} = \mathbf{b} - \mathbf{A}\hat{\mathbf{x}}$.
4: Solve $L\mathbf{t} = \mathbf{Pr}$ for $\mathbf{t}$.
5: Solve $U\mathbf{s} = \mathbf{t}$, for $\mathbf{s}$.
6: $\mathbf{x} = \hat{\mathbf{x}} + \mathbf{s}$.
7: $\hat{\mathbf{x}} = \mathbf{x}$.
8: **end for**

Output: $\mathbf{x}$.

what Golub and van Loan [23, pp. 124-127] has to say about the accuracy of computed solutions from Gaussian elimination. If the machine precision ε is such that $\varepsilon = 10^{-a}$, $a \in \mathbb{N}$ and the condition number of $\mathbf{A}$, $\kappa(\mathbf{A}) = 10^m$ where $m \in \mathbb{N}$, then Gaussian elimination produces a solution $\hat{\mathbf{x}}$ that has $(a - m)$ correct decimal digits.

If the working precision is double precision, then one of the ways of computing **r** accurately is by using quadruple precision (that is, double the working precision). That is, we compute **r** in quadruple precision before rounding it to double precision. This means that if 16-digit arithmetic is used to compute $\mathbf{PA} = LU$, **x**, **t** and **s** in Algorithm 5 [23, p. 127], then 32-digit arithmetic is used to compute $\mathbf{r} = \mathbf{b} - \mathbf{A}\hat{\mathbf{x}}$ in mixed precision iterative refinement, **P** is a permutation matrix. When mixed precision iterative refinement is used, Golub and van Loan [23, p. 127] make the following heuristic statement about the accuracy of the computed solution: if the machine precision $\varepsilon = 10^{-a}$ and $\kappa(\mathbf{A}) = 10^m$, then Algorithm 5 produces an **x** which has approximately $\min\{a, k(a - m)\}$ correct decimal digits. So $k = 1$ which corresponds to one iterative refinement shows no benefit according to Golub and van Loan.

Algorithm 5 Mixed Precision Iterative Refinement

Input: $\mathbf{A}, \hat{\mathbf{x}}, \mathbf{b}$

1: Compute LU factorization of $\mathbf{A}$.
2: **for** $k = 1, 2, \ldots,$ **do**
3: Compute the residual $\mathbf{r} = \mathbf{b} - \mathbf{A}\hat{\mathbf{x}}$, (using double the working precision).
4: Solve $L\mathbf{t} = \mathbf{Pr}$ for $\mathbf{t}$.
5: Solve $U\mathbf{s} = \mathbf{t}$, for $\mathbf{s}$.
6: $\mathbf{x} = \hat{\mathbf{x}} + \mathbf{s}$.
7: $\hat{\mathbf{x}} = \mathbf{x}$.
8: **end for**

Output: $\mathbf{x}$.

The above argument is based on the accuracy of the solution computed by fixed and mixed precision iterative refinements in which the solver for $\mathbf{A}$ is Gaussian elimination with partial pivoting. However, Higham [30, p. 234], gives the following two results on fixed and mixed precision iterative refinements.

Lemma 2.2.5. *(Fixed precision iterative refinement [30, p. 234]) Let $\mathbf{A}$ be an n by n nonsingular matrix and assume $\hat{L}$ and $\hat{U}$ are the computed LU factors of $\mathbf{A}$. Further, let fixed precision iterative refinement be applied to the linear system $\mathbf{Ax} = \mathbf{b}$, using LU factorisation. If ω defined by $\omega = \varepsilon \| \, |\mathbf{A}^{-1}||\hat{L}||\hat{U}| \, \|_\infty$ is sufficiently less than one, then iterative refinement reduces the error by a factor approximately ω at each stage until*

$$\frac{\|\mathbf{x} - \hat{\mathbf{x}}_k\|_\infty}{\|\mathbf{x}\|_\infty} \lesssim 2n\varepsilon \, cond(\mathbf{A}, \mathbf{x}),$$

where ε is machine precision, $|\mathbf{A}|$ is the componentwise absolute value of the elements in $\mathbf{A}$ and [30, p. 135]

$$cond(\mathbf{A}, \mathbf{x}) = \frac{\| \, |\mathbf{A}^{-1}||\mathbf{A}||\mathbf{x}| \, \|_\infty}{\|\mathbf{x}\|_\infty}.$$

Lemma 2.2.6. *(Mixed precision iterative refinement [30, p. 234]) Let iterative refinement be applied to the nonsingular linear system $\mathbf{Ax} = \mathbf{b}$, using LU factorisation and with residuals computed in quadruple precision. Let $\omega = \varepsilon \| |\mathbf{A}^{-1}||\hat{L}||\hat{U}| \|_\infty$, where $\hat{L}$ and $\hat{U}$ are the computed LU factors of $\mathbf{A}$. Then, provided ω is sufficiently less than one, iterative refinement reduces the error by a factor approximately ω at each stage*

until

$$\frac{\|\mathbf{x} - \hat{\mathbf{x}}\|_\infty}{\|\mathbf{x}\|_\infty} \approx \varepsilon.$$

The above result *i.e.,* Lemma 2.2.6 shows that Higham's result on mixed precision iterative refinement is at variance with Golub and van Loan's heuristic argument.

As stated earlier, because our input data are in double precision, in order to improve the accuracy of the computed solution in $\mathbf{A}\mathbf{x} = \mathbf{b}$, we would need to compute the residual in quadruple precision [15, p. 62] in a mixed precision iterative refinement. However, quadruple precision arithmetic may not be available or it takes ages to run. As a result of this, the numerical experiments that will be carried out in the next section are done in fixed precision iterative refinement with one step of iterative refinement. Of which, we will make use of Algorithm 4 to obtain numerical results.

In the next section, we present result of numerical experiments to support the theory developed so far.

2.3 Numerical Experiments

In this section, we discuss the performance of Algorithm 3 described earlier, on some numerical examples which confirms that quadratic convergence is achieved. In all numerical experiments, our aim is to find the particular value of γ^* such that two simple leftmost eigenvalues λ_1 and λ_2 of $\mathbf{A}(\gamma)$ coalesce at λ^* to form a 2-dimensional Jordan block. Typically, the two leftmost eigenvalues will be most important in applications.

The motivation for the examples we consider comes from the computation of fluid flows governed by the steady-state Navier-Stokes equations as presented in [28]. Assuming that a reference velocity field $\mathbf{v}$ has been computed for some particular parameter values. To assess its stability, it is necessary to solve the following partial differential equation eigenvalue problem [28, p. 1152]:

$$\begin{aligned} -\nu\Delta\mathbf{u} + \mathbf{v}.\nabla\mathbf{u} + \mathbf{u}.\nabla\mathbf{v} + \nabla p &= \lambda\mathbf{u} \\ \nabla.\mathbf{u} &= 0, \end{aligned} \tag{2.30}$$

where $\lambda \in \mathbb{C}$ is an eigenvalue and the pair $(\mathbf{u}, p)$ are the non-trivial eigenfunctions satisfying suitable homogeneous boundary conditions. The parameter ν is the viscosity, which is inversely proportional to the Reynolds number *Re*. The eigenfunction $(\mathbf{u}, p)$ consists of the velocity $\mathbf{u}$, and the pressure p.

For the numerical experiments discussed in this section, we modify (2.30) by assuming that the viscosity $\nu = 1$, the reference velocity $\mathbf{v}$ will be taken as $\mathbf{v} = [\gamma, \gamma]^T$ or $\mathbf{v} = [\gamma, 5]^T$, where γ is a constant over the domain, and the pressure term, ∇p is neglected. So that (2.30) reduces to

$$-\Delta \mathbf{u} + \gamma \mathbf{u}_x + \gamma \mathbf{u}_y = \lambda \mathbf{u}, \tag{2.31}$$

in Examples 2.3.2 and 2.3.4, and $-\Delta \mathbf{u} + \gamma \mathbf{u}_x + 5\mathbf{u}_y = \lambda \mathbf{u}$ in Example 2.3.3 respectively. The first example below explains the reason for choosing varying mesh sizes in the discretization of the partial differential equations eigenvalue problem in Examples 2.3.2, 2.3.3 and 2.3.4, below.

Example 2.3.1. *Consider the following parameter-dependent ordinary differential eigenvalue problem, discretized using finite centred differences (cf.* (2.31)*)*

$$-\frac{\mathrm{d}^2 u}{\mathrm{d}x^2} + \gamma \frac{\mathrm{d}u}{\mathrm{d}x} = \lambda u; \quad u(0) = u(1) = 0, \tag{2.32}$$

with a constant mesh size $h = \frac{1}{n+1}$. *Observe that for* $k = 1, 2, \ldots, n$ *we have the following discretized form of* (2.32)

$$\frac{-(h\gamma + 2)u_{k-1} + 4u_k + (h\gamma - 2)u_{k+1}}{2h^2} = \lambda u_k. \tag{2.33}$$

After imposing the initial conditions, the resulting discretized eigenvalue problem $\mathbf{A}(\gamma)\mathbf{u} = \lambda \mathbf{u}$ *is as follows*

$$\frac{1}{2h^2}\begin{bmatrix} 4 & (h\gamma-2) & & & & \\ -(h\gamma+2) & 4 & (h\gamma-2) & & & \\ & -(h\gamma+2) & 4 & (h\gamma-2) & & \\ & \cdots & \cdots & \cdots & \cdots & \\ & & & \cdots & (h\gamma-2) & \\ & & & -(h\gamma+2) & 4 & \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \cdots \\ u_n \end{bmatrix} = \lambda \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \cdots \\ u_n \end{bmatrix},$$

where $\mathbf{A}(\gamma)$ *is a n by n matrix and* $\mathbf{u} = [u_1, u_2, u_3, \ldots, u_n]^T$. *Note that for* $\gamma^* = \pm\frac{2}{h}$, $\mathbf{A}(\gamma^*)$ *has an* $n-$*dimensional Jordan block corresponding to the eigenvalue* $\lambda^* = 4$*-as*

against the Jordan block of dimension two we are interested in. The same phenomenon arises if one discretizes (2.31) *using equally spaced mesh sizes. As a result of this, in the following examples, we use a discretization with a variable mesh.*

Example 2.3.2. *Consider finding a 2-dimensional Jordan block of* $\mathbf{A}(\gamma)$*, derived by a finite centred difference discretization of the convection-diffusion eigenvalue problem*

$$\begin{aligned} -\Delta \mathbf{u} + \gamma \mathbf{u}_x + \gamma \mathbf{u}_y = \lambda \mathbf{u}, \quad \textit{in} \quad D := [0,1] \times [0,1], \\ u = 0, \quad \textit{on} \quad \Gamma := \partial D, \end{aligned} \tag{2.34}$$

on a 32 by 32 grid with 961 degrees of freedom[2]*. For* $\gamma = 5$ *and with a constant step size* $h = \frac{1}{32}$*, Golub and Ye [24] gave the two leftmost eigenvalues of* (2.34) *as* $\lambda_1 \approx 32.18$ *and* $\lambda_2 \approx 61.58$*, while the other eigenvalues satisfy* $Re(\lambda) \geq 61.58$*. However, for this particular numerical experiment, we will use variable mesh sizes, chosen as:* $h_1 = 0.15$*,* $h_2 = 0.2$*,* $h_3 = 0.3$*,* $h_{31} = 0.07$*,* $h_{32} = 0.09$*. The other intermediate values of h are computed by*

$$h_k = \frac{1 - (h_1 + h_2 + h_3 + h_{31} + h_{32})}{p - 4}, \quad \textit{for} \quad k = 4, 5, \cdots, 30, \quad p = 31.$$

The error tolerance in Algorithm 3 is taken to be 7×10^{-15}*. For* $\gamma^{(0)} = 15$*, we computed* $\mathbf{A}(\gamma^{(0)})$ *and applied two steps of inverse iteration on* $\mathbf{A}(\gamma^{(0)}) - \hat{\lambda}\mathbf{I}$ *with* $\hat{\lambda} = 72$ *as a shift. The initial starting vector for the inverse iteration is*

$$\mathbf{x}^{(0)} = [1, 0, 1, 0, 0, \cdots, 0, 0, 1, 0, 1]^T / 2.$$

We obtained the estimate $(\mathbf{x}, \lambda)$ *for the eigenpair, and took* $\lambda^{(0)} = \lambda$*. The vectors* $\mathbf{c}$ *and* $\mathbf{b}$ *were kept fixed as* $\mathbf{c} = \mathbf{x}$ *and* $\mathbf{b} = \mathbf{A}'(\gamma^{(0)})\mathbf{c}$*, as discussed in Section 2.2.1.*

The first two eigenvalues of $\mathbf{A}(\gamma^*)$ *coalesced at* 74.72506 *for* $\gamma^* = 15.17519$ *to form a 2-dimensional Jordan block as shown in Table 2.1. The solution is obtained by finding the LU factorization of* $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ *with one step of iterative refinement in each iteration. This became necessary so as to improve the accuracy of the computed solutions.*

Observe from the fifth and sixth columns of Table 2.2 that we obtain quadratic convergence for $k = 2, 3, 4$ *and 5. However, this quadratic convergence is lost from*

[2]Degrees of freedom is the number of interior mesh points.

k	$\lambda^{(k)}$	$\lvert\lambda^{(k+1)}-\lambda^{(k)}\rvert$	$\lvert\gamma^{(k+1)}-\gamma^{(k)}\rvert$	$\lVert\Delta\mathbf{y}^{(k)}\rVert$	$\lVert\mathbf{G}(\mathbf{y}^{(k)})\rVert$
0	72.00000	1.9e+00	8.7e-02	1.9e+00	4.9e-01
1	73.85590	7.7e-01	7.2e-02	7.7e-01	1.5e-01
2	74.62308	1.0e-01	1.6e-02	1.0e-01	2.3e-02
3	74.72358	1.5e-03	3.9e-04	1.5e-03	5.2e-04
4	74.72506	3.7e-07	1.3e-07	3.9e-07	1.7e-07
5	74.72506	1.8e-13	1.1e-14	1.9e-13	1.5e-14
6	74.72506	1.4e-14	0.0e+00	9.6e-15	8.9e-16
7	74.72506	0.0e+00	0.0e+00	1.5e-15	8.0e-16

Table 2.1: Values of $\gamma^{(k)}$ and $\lambda^{(k)}$ for the discretized convection-diffusion eigenvalue problem (2.34) in Example 2.3.2. Columns 5 and 6 show that the results converged quadratically for $k = 2, 3, 4$ and 5 as predicted by Theorem 2.2.2.

the sixth iterate because of round-off errors in computing the five unknowns and their respective residuals from (2.4), (2.6), (2.10), (2.13) *and* (2.14). *The condition number of* $\mathbf{M}(\lambda^*, \gamma^*)$ *is approximately* 2×10^8, *while that of* $\mathbf{G}_{\mathbf{y}}(\mathbf{y}^*)$ *is approximately 86. The large condition number of* $\mathbf{M}(\lambda^*, \gamma^*)$ *suggests that we will not achieve accuracy to full double precision, and this is indeed observed in the last three rows of the last two columns of Table 2.1. The computed nonzero values of* $f_\gamma(\lambda^*, \gamma^*)$ *and* $f_{\lambda\lambda}(\lambda^*, \gamma^*)$ *are approximately* -1.30221 *and* 0.01543 *respectively. So, the conditions of Theorems 2.2.1 and 2.2.2 are satisfied.*

Example 2.3.3. *This is the same with the first example but with the second* γ *replaced with a 5. Consider* $\mathbf{A}(\gamma)$ *derived by a finite centred difference discretization of the following convection-diffusion eigenvalue problem*

$$\begin{aligned} -\nabla^2\mathbf{u} + \gamma\mathbf{u}_x + 5\mathbf{u}_y = \lambda\mathbf{u}, \quad \textit{in} \quad D := [0,1]\times[0,1], \\ u = 0, \quad \textit{on} \quad \Gamma := \partial D, \end{aligned} \tag{2.35}$$

on a 32 by 32 grid with 961 degrees of freedom. This means that $\mathbf{A}(\gamma)$ *is of size 961 by 961. The error tolerance,* $tol = 5 \times 10^{-15}$ *while Table 2.2 shows the tabulated values of* λ *and* γ. *The solution is obtained by finding the LU factorization of* $\mathbf{M}(\lambda, \gamma)$ *plus one iterative refinement. For* $\gamma^{(0)} = -16$, *we computed* $\mathbf{A}(\gamma^{(0)})$ *and used two steps of inverse iteration with* $\hat{\lambda} = 47$ *as a shift to obtain an estimate for the eigenpair* $(\mathbf{x}, \lambda)$ *with* $\lambda^{(0)} = \lambda$, *and the initial starting vector for the inverse iteration is* $\mathbf{x}^{(0)} = [1, 0, 1, 0 \cdots, 0, 1, 0, 1]^T/2$. $\mathbf{c}$ *and* $\mathbf{b}$ *were kept fixed as* $\mathbf{c} = \mathbf{x}$, *the computed*

estimate of the eigenvector from inverse iteration and $\mathbf{b} = \mathbf{A}'(\gamma^{(0)})\mathbf{c}$. *The stopping criterion is as stated in* (2.18). *The variable step sizes are chosen as follows:* $h_1 = 0.15$, $h_2 = 0.22$, $h_3 = 0.41$, $h_{31} = 0.07$, $h_{32} = 0.1$, *the other intermediate values of h are computed by the relation*

$$h_k = \frac{1 - (h_1 + h_2 + h_3 + h_{31} + h_{32})}{p - 4}, \quad \textit{for} \quad k = 4, 5, \cdots, 30, \quad p = 31.$$

Quadratic convergence is easily seen from the fifth column of Table 2.2, from the fourth to the seventh iterates. The condition number of $\mathbf{M}(\lambda^*, \gamma^*)$ *and* $\mathbf{G}_{\mathbf{y}}(\mathbf{y}^*)$ *are approximately* 3×10^8 *and 14 respectively. In this example, which has a greater element of nonsymmetry than the previous example, we again fail to see convergence to machine precision, but given the size of the condition number of* $\mathbf{M}(\lambda, \gamma)$ *at the root, the results are remarkably good. From the numerical computations, we obtained* $f_\gamma(\lambda^*, \gamma^*) \approx -0.84065$, $f_{\lambda\lambda}(\lambda^*, \gamma^*) \approx 0.07262$, *which are nonzero. Thus, the computed values of* $f_\gamma(\lambda^*, \gamma^*)$ *and* $f_{\lambda\lambda}(\lambda^*, \gamma^*)$ *agree with the conditions of Theorems 2.2.1 and 2.2.2 that they are nonzero. From the numerical experiments, the first two*

k	$\lambda^{(k)}$	$\lvert\lambda^{(k+1)} - \lambda^{(k)}\rvert$	$\lvert\gamma^{(k+1)} - \gamma^{(k)}\rvert$	$\lVert\Delta\mathbf{y}^{(k)}\rVert$	$\lVert\mathbf{G}(\mathbf{y}^{(k)})\rVert$
0	47.99881	8.8e-01	1.8e-01	9.0e-01	9.8e-01
1	48.87667	1.5e+00	1.5e-02	1.5e+00	4.8e-01
2	50.40657	4.3e-01	2.1e-01	4.8e-01	1.9e-01
3	50.83488	4.3e-01	1.4e-02	4.3e-01	1.7e-02
4	50.40355	1.1e-01	1.5e-03	1.1e-01	1.1e-02
5	50.51520	1.0e-02	7.0e-04	1.0e-02	1.2e-03
6	50.52546	6.7e-05	7.6e-06	6.7e-05	1.0e-05
7	50.52553	2.0e-09	3.9e-10	2.1e-09	4.4e-10
8	50.52553	2.8e-14	1.8e-15	2.7e-14	2.9e-15
9	50.52553	0.0e+00	0.0e+00	1.0e-15	2.4e-16

Table 2.2: Values of $\gamma^{(k)}$ and $\lambda^{(k)}$ for the discretized convection-diffusion eigenvalue problem (2.35) in Example 2.3.3. Column five shows that the results converged quadratically as predicted by Theorem 2.2.2, with the exception of the last two rows.

real leftmost eigenvalues of $\mathbf{A}(\gamma^*)$ *coalesced at* $\lambda^* = 50.52553$ *for* $\gamma^* = -15.97019$.

Example 2.3.4. *This example is the same as that in Example 2.3.2 where* $\mathbf{A}(\gamma)$ *is derived by a finite difference discretization of the convection-diffusion eigenvalue prob-*

lem

$$-\nabla^2\mathbf{u} + \gamma\mathbf{u}_x + \gamma\mathbf{u}_y = \lambda\mathbf{u}, \quad in \quad D := [0,1]\times[0,1],$$
$$u = 0, \quad on \quad \Gamma := \partial D,$$

using centre differences but on a 64 by 64 grid so that there are 63×63 *(3969) degrees of freedom. The number of degrees of freedom in this example is approximately four times larger than that of Example 2.3.2. The variable step sizes are chosen as:* $h_1 = 0.15$, $h_2 = 0.2$, $h_3 = 0.3$, $h_{31} = 0.07$, $h_{32} = 0.09$. *The other intermediate values of h are computed by*

$$h_k = \frac{1-(h_1+h_2+h_3+h_{31}+h_{32})}{p-4}, \quad for \quad k = 4,5,\cdots,62, \quad p = 63.$$

The error tolerance is 8×10^{-15}. *Given* $\gamma^{(0)} = 0$, *we computed* $\mathbf{A}(\gamma^{(0)})$ *and used inverse iteration with* $\hat{\lambda} = 62$ *as a shift to obtain an estimate for the eigenpair* $(\mathbf{x},\lambda)$ *of -1 times the Laplacian, where* $\lambda^{(0)} = \lambda$. *The starting vector for the inverse iteration is* $\mathbf{x}^{(0)} = [1,0,0\cdots,0,0]^T$. $\mathbf{c}$ *and* $\mathbf{b}$ *were kept fixed as* $\mathbf{c} = \mathbf{x}$ *and* $\mathbf{b} = \mathbf{A}'(\gamma^{(0)})\mathbf{c}$. *The first two real eigenvalues of* $\mathbf{A}(\gamma^*)$ *coalesced at* $\lambda^* = 63.33495$ *for* $\gamma^* = -0.25951$ *to form a 2-dimensional Jordan block as shown in Table 2.3. The solution is obtained by finding the LU factorization of* $\mathbf{M}(\lambda^{(k)},\gamma^{(k)})$ *plus one iterative refinement in each iteration. We observed quadratic convergence for* $k = 2,3,4$ *and* 5 *of column five. The loss of quadratic convergence in the last two rows of Table 2.3 is due to the imprecision in computing the residual in each of the linear solves of Algorithm 3 as stated in Lemma 2.2.5 in the previous section. At the root,* $f_\gamma(\lambda,\gamma) = -0.75971$ *and* $f_{\lambda\lambda}(\lambda,\gamma) = -0.32598$, *so the conditions of Theorem 2.2.1 and 2.2.2 are satisfied.*

Though no numerical computation is done, the next example is a potential application of the theory discussed in this chapter.

Example 2.3.5. *Consider the following generalized eigenvalue problem*

$$(\mathbf{K}_T + \gamma\mathbf{A})\mathbf{q} = \lambda\mathbf{M}\mathbf{q}, \tag{2.36}$$

arising from the finite element discretization of the supersonic panel flutter problem (as discussed in the third paragraph of Chapter 1), where $\mathbf{K}_T$ *and* $\mathbf{M}$ *are the total stiffness and consistent mass matrices respectively which are symmetric positive definite,*

k	$\lambda^{(k)}$	$\lvert\lambda^{(k+1)} - \lambda^{(k)}\rvert$	$\lvert\gamma^{(k+1)} - \gamma^{(k)}\rvert$	$\lVert\Delta\mathbf{y}^{(k)}\rVert$	$\lVert\mathbf{G}(\mathbf{y}^{(k)})\rVert$
0	62.00001	6.8e-01	1.8e-01	7.1e-01	6.1e-01
1	62.68194	5.6e-01	2.3e-02	5.6e-01	2.1e-01
2	63.24539	9.2e-02	5.5e-02	1.1e-01	4.7e-02
3	63.33725	2.3e-03	8.2e-04	2.4e-03	1.1e-03
4	63.33494	3.4e-06	1.6e-06	3.8e-06	1.5e-06
5	63.33495	6.1e-12	1.2e-12	6.3e-12	2.4e-12
6	63.33495	0.0e+00	2.0e-14	2.1e-14	1.6e-14
7	63.33495	0.0e+00	2.8e-16	6.7e-16	3.2e-16

Table 2.3: Values of $\gamma^{(k)}$ and $\lambda^{(k)}$ for the discretized convection-diffusion eigenvalue problem (2.34) in Example 2.3.4. Column 5 show that, for $k = 2, 3, 4$ and 5 we obtained quadratic convergence as predicted by Theorem 2.2.2.

and $\mathbf{A}$ *is the nonsymmetric aerodynamic load matrix. In this context,* γ *represents the dynamic pressure parameter and the pair* $\mathbf{q}$ *and* λ *represent displacements and eigenvalues respectively. When* $\gamma = 0$*, the eigenvalues of* (2.36) *are real and positive. However, as* γ *is increased monotonically from zero, the first two smallest eigenvalues* λ_1 *and* λ_2*, move and coalesce together to* λ^* *at* $\gamma = \gamma^*$ *and become complex conjugate eigenpairs (see, for example, [43, pp. 2268-2269], [44, p. 748]). By making the substitutions* $\mathbf{x} = L^T\mathbf{q}$ *and* $\mathbf{M} = LL^T$*, the above generalized eigenvalue problem reduces to*

$$(\mathbf{B} + \gamma\mathbf{C})\mathbf{x} = \lambda\mathbf{x}, \tag{2.37}$$

where $\mathbf{B} = L^{-1}\mathbf{K}_T L^{-T}$*,* $\mathbf{C} = L^{-1}\mathbf{A}L^{-T}$ *and in the notation of this chapter* (2.37) *becomes* $\mathbf{A}(\gamma)\mathbf{x} = \lambda\mathbf{x}$ *with* $\mathbf{A}(\gamma) = \mathbf{B} + \gamma\mathbf{C}$*. Note that* $\mathbf{M} = LL^T$ *is the Cholesky factorization [52, p. 262] of* $\mathbf{M}$ *where* L *is a lower triangular matrix with positive diagonal elements.*

For us to solve the five bordered linear system of equations in Algorithm 3 efficiently, in the next section, we present a description of the Block Elimination Mixed method (BEM) algorithm.

2.4 Efficient Solves using Block Elimination Mixed Method

This section is motivated by a desire to solve the bordered linear system of equations (2.4), (2.6), (2.10), (2.13), (2.14) for $f(\lambda,\gamma)$, $f_\lambda(\lambda,\gamma)$, $f_{\lambda\lambda}(\lambda,\gamma)$, $f_\gamma(\lambda,\gamma)$, $f_{\lambda\gamma}(\lambda,\gamma)$ respectively in Algorithm 3 more efficiently when $\mathbf{A}$ arises from a discretized partial differential equation eigenvalue problem and hence has special structure. This is accomplished using the Block Elimination Mixed method (BEM) of Govaerts and Pryce [25], [27] for solving bordered linear system of equations. There are three subsections. In Subsection 2.4.1, we present the Block Elimination Doolittle (BED) and the Block Elimination Crout (BEC) algorithm. In Subsection 2.4.2, we describe the Block Elimination Mixed (BEM) algorithm. A description of the Thomas algorithm for solving block tridiagonal systems is given in Subsection 2.4.3. We will compare the computational time of solving all the systems involving $\mathbf{M}(\lambda,\gamma)$ on the left hand side using LU-factorization with the time it takes using BEM in Examples 2.3.2, 2.3.3 and 2.3.4 of the last section.

First, we give a brief discussion of what is meant by a stable solver. A solver $\mathbf{T}$ for $\mathbf{A}$ is a map $\mathbf{T}:\mathbb{R}^n\to\mathbb{R}^n$ such that $\mathbf{T}(\mathbf{v})$ is an approximate solution to $\mathbf{AT}=\mathbf{v}$. $\mathbf{T}$ is stable if there exists $\Delta\mathbf{A}$, $\Delta\mathbf{v}$ such that [25, p. 470]

$$(\mathbf{A}+\Delta\mathbf{A})\mathbf{T}(\mathbf{v})=\mathbf{v}+\Delta\mathbf{v},\quad \text{where}\quad \|\Delta\mathbf{A}\|\le\varepsilon C_{\mathbf{T}}\|\mathbf{A}\|,\ \ \|\Delta\mathbf{v}\|\le\varepsilon C_{\mathbf{T}}\|\mathbf{v}\|, \tag{2.38}$$

ε is a floating point round off unit and $C_{\mathbf{T}}$ is a stability constant of $\mathbf{T}$ which is modest [26, p. 497, 500]. Examples of stable solvers are: QR factorization, Cholesky factorization, while an example of an unstable solver is Gaussian elimination without pivoting. However, following [60, pp. 163-170], "Gaussian elimination with partial pivoting is explosively unstable for certain matrices, yet stable in practice." It is unstable for those matrices in which the growth factor[3] is large. In this thesis, we shall take the view of [60] and assume that Gaussian elimination with partial pivoting is a stable solver. When we used

[3]Assuming $\mathbf{A}$ has been factored into $\mathbf{PA}=LU$, then the growth factor of $\mathbf{A}$ is defined as $\xi=\frac{\max_{i,j}|u_{ij}|}{\max_{i,j}|a_{ij}|}$, where u_{ij} and a_{ij} are the entries of U and $\mathbf{A}$ respectively. Typically for stability, ξ should be of order one [60, p. 165].

QR factorization in the numerical examples of the last section, there was no significant improvements than those obtained using LU.

Let $\mathbf{A}$ be singular or nearly singular and consider the problem of solving the following linear system of equations $\mathbf{Mp} = \mathbf{b}$,

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ e \end{bmatrix}, \tag{2.39}$$

with $f, e \in \mathbb{R}$ and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, where

$$\mathbf{M} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)},$$

is nonsingular. One of the ways of solving (2.39) is to apply the LU factorization directly to $\mathbf{M}$ but this will probably destroy any structure in $\mathbf{A}$. However, BEM (Block Elimination Mixed) is an alternative method of obtaining $\mathbf{x}$ and f which is more efficient and takes advantage of the structure in $\mathbf{A}$. BEM is based on two block elimination methods: Block Elimination Doolittle (BED) and Block Elimination Crout (BEC). Following [25], if $\mathbf{M}$ is well conditioned and $\mathbf{A}$, $\mathbf{A}^T$ are solved in a stable way, then BED produces f accurately without any iterative refinement. In the same vein, BEC produces $\mathbf{x}$ accurately. The backward error analysis shows that the accuracy of the solution obtained by BEC depends on the size of the norm of the computed $\mathbf{h}$-which is a vector that is defined later in Algorithms 7 and 8 (see, [25, Proposition 3.3, p. 476-477]). In a nutshell, BEM uses the 'relative strengths' of the two algorithms. This forms the basis for the following discussion on BED and BEC.

2.4.1 Block Elimination Doolittle (BED) and Block Elimination Crout (BEC)

BED is a method of solving the bordered linear system of equations (2.39) by factoring $\mathbf{M}$ blockwise using the following Doolittle factorization (see, for ex-

ample, [27, p. 162], [25, p. 470])

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{w}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0}^T & \mu_1 \end{bmatrix},$$

so that after expanding the right hand side

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{w}^T\mathbf{A} & \mathbf{w}^T\mathbf{b} + \mu_1 \end{bmatrix},$$

and equating to the left hand side, we obtain $\mathbf{w}^T\mathbf{A} = \mathbf{c}^T$ which implies

$$\mathbf{A}^T\mathbf{w} = \mathbf{c}.$$

In the same vein, since $d = \mathbf{w}^T\mathbf{b} + \mu_1$, we have

$$\mu_1 = d - \mathbf{w}^T\mathbf{b}.$$

Note that μ_1 is not equal to zero because $\mathbf{M}$ is assumed nonsingular. Moreover, by expanding along the last row of

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{w}^T\mathbf{A} & \mathbf{w}^T\mathbf{b} + \mu_1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ e \end{bmatrix},$$

we obtain $\mathbf{w}^T\mathbf{A}\mathbf{x} + f\mathbf{w}^T\mathbf{b} + f\mu_1 = e$ and $\mathbf{w}^T(\mathbf{A}\mathbf{x} + f\mathbf{b}) + f\mu_1 = e$. Using $\mathbf{A}\mathbf{x} + f\mathbf{b} = \mathbf{y}$, yields $\mathbf{w}^T\mathbf{y} + f\mu_1 = e$. Which implies

$$f = (e - \mathbf{w}^T\mathbf{y})/\mu_1.$$

The above analysis now gives rise to Algorithm 6 (see, for example, [25]).

Note that the first two steps in Algorithm 6 compute the block Doolittle factorization and do not rely on the right hand side of (2.39). Furthermore, the backward error analysis shows that the computed f is an exact solution of a system near $\mathbf{M}\mathbf{p} = \mathbf{g}$ (see, for example, [25, pp. 474-475]), hence, BED gives accurate approximations to f without any iterative refinement. However, it was observed numerically that one major drawback of BED [25, p. 471] is that

Algorithm 6 BED Algorithm for solving Bordered Linear Systems

Input: $\mathbf{M}$ partitioned into $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$, d and right hand sides $\mathbf{y}$ and e.

1: Solve $\mathbf{A}^T\mathbf{w} = \mathbf{c}$ for $\mathbf{w}$.
2: Compute $\mu_1 = d - \mathbf{w}^T\mathbf{b}$.
3: Compute $f = (e - \mathbf{w}^T\mathbf{y})/\mu_1$.
4: Solve $\mathbf{A}\mathbf{x} = \mathbf{y} - f\mathbf{b}$ for $\mathbf{x}$.

Output: $\mathbf{x}$, f.

more iterative refinement steps are needed to give a good approximation to $\mathbf{x}$. Next, we present a description of the BEC algorithm.

BEC is another way of solving (2.39) by using the following block Crout factorization (see, for example [27, p. 162], [25, p. 470])

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{c}^T & \mu \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{q} \\ \mathbf{0}^T & 1 \end{bmatrix}, \quad \mu \neq 0.$$

After expanding the right hand side,

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{A}\mathbf{q} \\ \mathbf{c}^T & \mathbf{c}^T\mathbf{q} + \mu \end{bmatrix}.$$

By equating both sides componentwise, we obtain

$$\mathbf{A}\mathbf{q} = \mathbf{b},$$

and

$$\mu = d - \mathbf{c}^T\mathbf{q}.$$

Observe that because $\mathbf{M}$ is nonsingular by assumption, this means that μ cannot be zero. If we expand along the first n rows of

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{A}\mathbf{q} \\ \mathbf{c}^T & \mathbf{c}^T\mathbf{q} + \mu \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ e \end{bmatrix}, \tag{2.40}$$

we obtain $\mathbf{A}\mathbf{x} + f\mathbf{A}\mathbf{q} = \mathbf{y}$. By letting $\mathbf{h} = \mathbf{x} + f\mathbf{q}$, $\mathbf{x} = \mathbf{h} - f\mathbf{q}$, implies

$$\mathbf{A}(\mathbf{x} + f\mathbf{q}) = \mathbf{y}$$

can be rewritten as

$$\mathbf{Ah} = \mathbf{y}.$$

Now observe that from the last row of (2.40), $\mathbf{c}^T\mathbf{x} + f(\mathbf{c}^T\mathbf{q} + \mu) = e$. After some simplifications using $\mathbf{h} = \mathbf{x} + f\mathbf{q}$, we obtain

$$f = (e - \mathbf{c}^T\mathbf{h})/\mu.$$

These now give rise to the following algorithm: BEC Algorithm 7 ([25, p. 470], [27, pp. 162-163]). The two algorithms outlined above, BED and BEC give

Algorithm 7 BEC Algorithm for solving Bordered Linear Systems

Input: $\mathbf{M}$ partitioned into $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$, d and right hand sides $\mathbf{y}$ and e.

1: Solve $\mathbf{Aq} = \mathbf{b}$ for $\mathbf{q}$.
2: Compute $\mu = d - \mathbf{c}^T\mathbf{q}$.
3: Solve $\mathbf{Ah} = \mathbf{y}$ for $\mathbf{h}$.
4: Compute $f = (e - \mathbf{c}^T\mathbf{h})/\mu$.
5: Compute $\mathbf{x} = \mathbf{h} - f\mathbf{q}$ for $\mathbf{x}$.

Output: $\mathbf{x}$, f.

very good approximations to both $\mathbf{x}$ and f if $\mathbf{M}$ and $\mathbf{A}$ are well conditioned or $\mathbf{M}$ is well conditioned and $\mathbf{A}$ is not too ill-conditioned and a solver for $\mathbf{A}$ and $\mathbf{A}^T$ is stable [25, p. 470]. However, Govaerts and Pryce ([25, p. 470] [26, 492-493]) suggested that, in the special case when $\mathbf{A}$ is less well conditioned, the computed solutions can be improved by iterative refinement. Govaerts and Pryce [26, 492-493] used LU and QR solvers in showing that BEC+1[4] (BEC plus one iterative refinement) gives $\mathbf{x}$ and f accurately with two examples in which $\mathbf{A}$ is singular in one and nearly singular in the other. Nevertheless, it has been shown numerically in [27, 471-472] that BEC+1 fails (failure in computing $\mathbf{x}$ accurately) with a solver based on the preconditioned conjugate gradient algorithm applied to a symmetric positive-semidefinite $\mathbf{A}$ (with the diagonals of $\mathbf{A}$ as preconditioner) and proposed the method of Block Elimination Mixed (BEM) ([26, p. 491], [25, p. 470], [27, pp. 162-163]).

[4] In the iterative refinement step of BEC+1, we do not recompute $\mathbf{q}$ and μ again *i.e.*, Crout's factorization is done once. Meaning that BEC+1 entails only 3 "black box" solves for $\mathbf{A}$ [26, p. 492].

2.4.2 Block Elimination Mixed method (BEM)

In this section, we present the Block Elimination Mixed method (BEM) algorithm given by Govaerts [25] which combines the two previous algorithms.

BEM uses a mixture of BED and BEC in which f is first computed by BED, the value of f obtained in addition to a zero vector as an approximate solution to $\mathbf{x}$, are then used in one step of BEC [25, p. 472] to produce the following algorithm: BEM Algorithm 8.

Algorithm 8 BEM Algorithm for Solving Bordered Linear Systems

Input: $\mathbf{M}$ partitioned into $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$, d and right hand sides $\mathbf{y}$ and e.

1: Solve $\mathbf{A}^T\mathbf{w} = \mathbf{c}$ for $\mathbf{w}$.
2: Calculate $\mu_1 = d - \mathbf{w}^T\mathbf{b}$.
3: Calculate $f = (e - \mathbf{w}^T\mathbf{y})/\mu_1$.
4: Solve $\mathbf{A}\mathbf{q} = \mathbf{b}$ for $\mathbf{q}$.
5: Calculate $\mu = d - \mathbf{c}^T\mathbf{q}$.
6: Compute $\mathbf{y}_1 = \mathbf{y} - f\mathbf{b}$.
7: Compute $e_1 = e - fd$.
8: Solve $\mathbf{A}\mathbf{h} = \mathbf{y}_1$ for $\mathbf{h}$.
9: Calculate $f_1 = (e_1 - \mathbf{c}^T\mathbf{h})/\mu$.
10: Compute $\mathbf{x} = \mathbf{h} - f_1\mathbf{q}$.
11: Compute $f = f + f_1$.

Output: $\mathbf{x}$, f.

The error analysis of BEM in [25] shows that it gives accurate approximations to both $\mathbf{x}$ and f if $\mathbf{M}$ is well conditioned and the solvers for both $\mathbf{A}$ and $\mathbf{A}^T$ are stable. Observe from Algorithm 8, that the first three steps are the preprocessing part of BED, steps 4-5 of BEM are the preprocessing part of BEC. Furthermore, steps 6-7 compute the residuals with the f obtained in step 3 and a zero vector for $\mathbf{x}$ as first approximations to the solution. Steps 8-10 corresponds to the residual correction by BEC, while in step 11, the computed value of f is updated. For a solver based on 'LU' factorization, only one LU factorization is needed to implement steps 1, 4 and 8 of BEM. This is because $\mathbf{A} = LU$ implies $\mathbf{A}^T = U^TL^T$. It should be remarked that step 4 of BED is omitted in BEM to escape another solve for $\mathbf{A}$. Had we included step 4 of BED in BEM, we then modify steps 6-7 and update $\mathbf{x} = \mathbf{x} + \mathbf{x}_1$ in step 12 [25, p. 473].

If $\mathbf{A}$ is ill-conditioned, then the computed results of BEM can be improved by one (*i.e.*, BEM+1) or more iterative refinements. BEM+1 requires just four

solves, namely, the three solves in Algorithm 8 plus one solve in step 7 of Algorithm 9. Next, we present the iterative refinement steps of BEM: Algorithm 9.

Algorithm 9 BEM+k Algorithm (k number of iterative refinements)

Input: $\mathbf{M}$ partitioned into $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$, d and right hand sides $\mathbf{y}$ and e.

1: Use BEM Algorithm 8 to compute $\mathbf{x}_1$ and f_1, $\mathbf{w}$, $\mathbf{q}$, μ_1 and μ.
2: **for** $k = 1, 2, \ldots,$ **do**
3: Compute the residuals
$\mathbf{y}_1 = \mathbf{y} - \mathbf{A}\mathbf{x}_1 - f_1\mathbf{b}$.
$e_1 = e - \mathbf{c}^T\mathbf{x}_1 - f_1 d$.
4: Calculate $f = (e_1 - \mathbf{w}^T\mathbf{y}_1)/\mu_1$.
5: Compute $\mathbf{y}_2 = \mathbf{y}_1 - f\mathbf{b}$.
6: Compute $e_2 = e_1 - fd$.
7: Solve $\mathbf{A}\mathbf{h} = \mathbf{y}_2$ for $\mathbf{h}$ using Thomas algorithm or any other solver.
8: Calculate $f_2 = (e_2 - \mathbf{c}^T\mathbf{h})/\mu$.
9: Compute $\mathbf{x}_2 = \mathbf{h} - f_2\mathbf{q}$.
10: Compute $f_2 = f + f_2$.
11: Update $\mathbf{x}_1 = \mathbf{x}_1 + \mathbf{x}_2$ and $f_1 = f_1 + f_2$.
12: **end for**

Output: $\mathbf{x}_1$, f_1.

We conclude this section by stating that given stable solvers for $\mathbf{A}$ and $\mathbf{A}^T$, BEM is a stable solver. Let C_T be the common upper bound of the stability constants of the solvers with $\mathbf{A}$ and $\mathbf{A}$ transposed, The stability result in [27, p. 165] shows that provided $\mathbf{M}$ is not too ill-conditioned, the stability constant of BEM is a linear combination of C_T and the inner product constant [5] C_P.

Often in cases where the matrix $\mathbf{A}$ arises from the five-point finite difference discretization of partial differential equations, it usually has special structure, that is, large sparse and block tridiagonal. In which an 'LU' type factorization destroys the sparsity structure within the blocks, it is better to use solvers that preserve this block structure. One of such structure preserving direct solver is the block Thomas algorithm. In the next section, we describe the block Thomas algorithm for solving block tridiagonal systems.

[5] The inner product constant C_p is such that $fl(\mathbf{x}^T\mathbf{y}) = \mathbf{x}^T\vartheta\mathbf{y}$, $\vartheta \in (\varepsilon C_P)$ where ϑ is a diagonal matrix and $C_P \leq n$, n is the size of $\mathbf{A}$. In the scalar case, the notation $\vartheta \in 1(\delta)$, $\delta > 0$ means $\vartheta = e^{\mu}$ where $|\mu| \leq \delta$. In the matrix case, ϑ is a product of a finite number of matrices $\exp(E_k)$ where $\sum_k \|E_k\| \leq \delta$ [27, p. 474].

2.4.3 Thomas Algorithm for Solving Block Tridiagonal Systems

We consider the problem of solving a square linear system of equations, $\mathbf{Ax} = \mathbf{b}$ in which $\mathbf{A}$ is large sparse, partitioned into blocks, each block is of size N by N and is either diagonal or tridiagonal. In practical applications, such matrices arise from a five-point finite difference discretization of partial differential equations as in Examples 2.3.2, 2.3.3 and 2.3.4.

This section is structured as follows, we use a block LU-type factorization in factoring $\mathbf{A}$, after which block forward and backward substitutions are used in solving for the unknown vector-$\mathbf{x}$. We present Algorithm 10, which is actually the Thomas algorithm for solving block tridiagonal systems, and the computational cost in terms of the number of floating point operations required. The material in this section can be found in [31, pp. 58-61], [20, pp. 121-122] and [1].

Let $\mathbf{A}$ be partitioned as

$$\mathbf{A} = \begin{bmatrix} \mathbf{B}_1 & \mathbf{C}_1 & & & & \\ \mathbf{A}_2 & \mathbf{B}_2 & \mathbf{C}_2 & & & \\ & \mathbf{A}_3 & \mathbf{B}_3 & \mathbf{C}_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & \mathbf{A}_{N-1} & \mathbf{B}_{N-1} & \mathbf{C}_{N-1} \\ & & & & \mathbf{A}_N & \mathbf{B}_N \end{bmatrix}, \tag{2.41}$$

where the $\mathbf{A}_k, \mathbf{B}_k, \mathbf{C}_k$'s are of size N by N. Note that the $\mathbf{A}_k, \mathbf{B}_k, \mathbf{C}_k$'s do not have to be equal. The unknown vector $\mathbf{x}$ and corresponding right hand side is partitioned as

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}, \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \\ \vdots \\ \mathbf{b}_N \end{bmatrix}, \tag{2.42}$$

where each $\mathbf{x}_k$ and $\mathbf{b}_k$ are in $\mathbb{R}^N$. We factor $\mathbf{A}$ into a block LU type factorization

of the form

$$\mathbf{A} = LU = \begin{bmatrix} \boldsymbol{\Gamma}_1 & & & & & \\ \boldsymbol{\Theta}_2 & \boldsymbol{\Gamma}_2 & & & & \\ & \boldsymbol{\Theta}_3 & \boldsymbol{\Gamma}_3 & & & \\ & & \ddots & \ddots & & \\ & & & \boldsymbol{\Theta}_{N-1} & \boldsymbol{\Gamma}_{N-1} & \\ & & & & \boldsymbol{\Theta}_N & \boldsymbol{\Gamma}_N \end{bmatrix} \begin{bmatrix} \mathbf{I} & \boldsymbol{\Delta}_1 & & & & \\ & \mathbf{I} & \boldsymbol{\Delta}_2 & & & \\ & & \mathbf{I} & \boldsymbol{\Delta}_3 & & \\ & & & \ddots & \ddots & \\ & & & & \mathbf{I} & \boldsymbol{\Delta}_{N-1} \\ & & & & & \mathbf{I} \end{bmatrix}, \tag{2.43}$$

where $\mathbf{I}$ is the N by N identity matrix, $\boldsymbol{\Theta}_k$, $\boldsymbol{\Delta}_k$ and $\boldsymbol{\Gamma}_k$ are square matrices. The 'L' and 'U' factors are block bidiagonal and the above factorization is not unique. This is because, we can also factor $\mathbf{A}$ as

$$\mathbf{A} = \begin{bmatrix} \mathbf{I} & & & & & \\ \boldsymbol{\Delta}_1 & \mathbf{I} & & & & \\ & \boldsymbol{\Delta}_2 & \mathbf{I} & & & \\ & & \ddots & \ddots & & \\ & & & \boldsymbol{\Delta}_{N-2} & \mathbf{I} & \\ & & & & \boldsymbol{\Delta}_{N-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Gamma}_1 & \boldsymbol{\Theta}_2 & & & & \\ & \boldsymbol{\Gamma}_2 & \boldsymbol{\Theta}_3 & & & \\ & & \boldsymbol{\Gamma}_3 & \boldsymbol{\Theta}_4 & & \\ & & & \ddots & \ddots & \\ & & & & \boldsymbol{\Gamma}_{N-1} & \boldsymbol{\Theta}_N \\ & & & & & \boldsymbol{\Gamma}_N \end{bmatrix}.$$

After expanding the right hand side of (2.43) and comparing with the entries of $\mathbf{A}$ in (2.41) blockwisely, we obtain

$$\boldsymbol{\Theta}_k = \mathbf{A}_k, \quad \text{for} \quad k = 2, 3, \ldots, N,$$

$$\boldsymbol{\Gamma}_1 = \mathbf{B}_1, \quad \text{and} \quad \boldsymbol{\Gamma}_1 \boldsymbol{\Delta}_1 = \mathbf{C}_1,$$

and the following recurrence $\boldsymbol{\Gamma}_k \Delta_k = \mathbf{C}_k$, where

$$\boldsymbol{\Gamma}_k = \mathbf{B}_k - \mathbf{A}_k \boldsymbol{\Delta}_{k-1},$$

for $k = 2, 3, \ldots N$. We first solve for $\boldsymbol{\Delta}_k$, use the previous $\boldsymbol{\Delta}_{k-1}$ and then substitute into $\boldsymbol{\Gamma}_k = \mathbf{B}_k - \mathbf{A}_k \boldsymbol{\Delta}_{k-1}$ to get the $\boldsymbol{\Gamma}_k$'s. This completes the block LU factorization of $\mathbf{A}$. The system $\mathbf{A}\mathbf{x} = \mathbf{b}$ now reduces to solving

$$L\mathbf{y} = \mathbf{b}, \quad \text{and} \quad U\mathbf{x} = \mathbf{y},$$

for $\mathbf{y}$ and $\mathbf{x}$ respectively. Now, using the L factor in (2.42), we can rewrite $L\mathbf{y} = \mathbf{b}$ as

$$L\mathbf{y} = \begin{bmatrix} \boldsymbol{\Gamma}_1 & & & & & & \\ \mathbf{A}_2 & \boldsymbol{\Gamma}_2 & & & & & \\ & \mathbf{A}_3 & \boldsymbol{\Gamma}_3 & & & & \\ & & \ddots & \ddots & & & \\ & & & \mathbf{A}_{N-1} & \boldsymbol{\Gamma}_{N-1} & & \\ & & & & \mathbf{A}_N & \boldsymbol{\Gamma}_N \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \vdots \\ \mathbf{y}_{N-1} \\ \mathbf{y}_N \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \\ \vdots \\ \mathbf{b}_{N-1} \\ \mathbf{b}_N \end{bmatrix}.$$

Observe that one can solve for the $\mathbf{y}_k$'s, using forward substitution, beginning with $\boldsymbol{\Gamma}_1\mathbf{y}_1 = \mathbf{b}_1$, for $\mathbf{y}_1$. Then for $k = 2, 3, \ldots, N$, we solve for the remaining $\mathbf{y}_k$'s from the relation

$$\boldsymbol{\Gamma}_k\mathbf{y}_k = \mathbf{b}_k - \mathbf{A}_k\mathbf{y}_{k-1}.$$

We now substitute the computed values of $\mathbf{y}_k$ into $U\mathbf{x} = \mathbf{y}$, that is,

$$\begin{bmatrix} \mathbf{I} & \boldsymbol{\Delta}_1 & & & & & \\ & \mathbf{I} & \boldsymbol{\Delta}_2 & & & & \\ & & \mathbf{I} & \boldsymbol{\Delta}_3 & & & \\ & & & \ddots & \ddots & & \\ & & & & \mathbf{I} & \boldsymbol{\Delta}_{N-1} & \\ & & & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_{N-1} \\ \mathbf{x}_N \end{bmatrix} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \vdots \\ \mathbf{y}_{N-1} \\ \mathbf{y}_N \end{bmatrix}.$$

It is easily seen from the last row above that $\mathbf{x}_N = \mathbf{y}_N$. Using backward substitution, we obtain the remaining $\mathbf{x}_k$'s from the recurrence relation

$$\mathbf{x}_k = \mathbf{y}_k - \boldsymbol{\Delta}_k\mathbf{x}_{k+1},$$

for $k = N-1, N-2, \ldots, 2, 1$. The above theory now leads to Algorithm 10: Thomas Algorithm (see, for example, [31, pp. 58-61] and [20, pp. 121-122]). Note that 'LU' factorization can be used in the four block solves in Algorithm 10. This is what makes the Thomas algorithm an efficient solver for BEM, because one can store the 'L' and 'U' factors for $\boldsymbol{\Gamma}_1$ and $\boldsymbol{\Gamma}_k$ in steps 2 and 6 of Algorithm 10 to solve the systems in steps 1, 4 and 8 of BEM Algorithm 8.

Next, we briefly describe the operation counts for the Thomas Algorithm.

Algorithm 10 Thomas Algorithm for block Tridiagonal Systems

Input: $\mathbf{A}$, $\mathbf{b}$ and N.

1: Set $\Gamma_1 = \mathbf{B}_1$.
2: Solve $\mathbf{\Gamma}_1\mathbf{\Delta}_1 = \mathbf{C}_1$ for $\mathbf{\Delta}_1$.
3: Solve $\mathbf{\Gamma}_1\mathbf{y}_1 = \mathbf{b}_1$ for $\mathbf{y}_1$.
4: **for** $k = 2, 3 \ldots, N$ **do**
5: Compute $\mathbf{\Gamma}_k = \mathbf{B}_k - \mathbf{A}_k\mathbf{\Delta}_{k-1}$.
6: Solve $\mathbf{\Gamma}_k\mathbf{\Delta}_k = \mathbf{C}_k$ for $\mathbf{\Delta}_k$.
7: Solve $\mathbf{\Gamma}_k\mathbf{y}_k = \mathbf{b}_k - \mathbf{A}_k\mathbf{y}_{k-1}$, for $\mathbf{y}_k$.
8: **end for**
9: Set $\mathbf{x}_N = \mathbf{y}_N$.
10: **for** $k = N-1 : -1 : 1$ **do**
11: $\mathbf{x}_k = \mathbf{y}_k - \mathbf{\Delta}_k\mathbf{x}_{k+1}$.
12: **end for**

Output: $\mathbf{x}$

Note that since $\mathbf{A}$ has been partitioned into N blocks in N unknowns, this means that there are $n = N^2$ unknowns. Observe that the solution of $\mathbf{Ax} = \mathbf{b}$ by LU factorization costs $O(N^2)^3$ operations. Since the product of two N by N matrices requires N^3 operations [31], this implies that forward substitution which involves $(N-1)$ matrix-matrix multiplication at a cost of $2N^3$ operations and approximately $2(N^2)^2$ total operations. Similarly, there are $(N-1)$ LU factorizations at $\frac{2}{3}N^3$ operations which amounts to approximately $O(N^2)^2$ operations. Moreover, N^2 triangular solves are required at a cost of N^2 operations and $\sim O(N^2)^2$ floating point operations. Therefore, the total floating point operations required for Thomas block tridiagonal algorithm is approximately $O(N^2)^2$ operations. Hence, in solving $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A}$ is large and sparse, Thomas algorithm requires less number of operations and storage than the direct LU.

Finally, we compare the computational time and number of iterations obtained by using direct LU factorization to solve the bordered linear systems of Algorithm 3 and BEM on three numerical examples from the last section. In all numerical examples, the solver for BEM is the Thomas algorithm for block tridiagonal systems. The comparison table is drawn based on the results presented in Tables 2.1, 2.2 and 2.3 of Examples 2.3.2, 2.3.3 and 2.3.4 respectively.

Example 2.4.1. *We repeated the computations in Section 2.3 using BEM with the*

same initial guesses and stopping tolerance. Table 2.4 gives a comparison of the cpu time and number of iterations obtained by solving the five bordered systems in Algorithm 2.2.2 using LU factorization of $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ *and BEM. In using BEM, at each stage of the iteration, we used the Thomas algorithm in solving the three linear systems in BEM involving* $\mathbf{A}(\gamma^{(k)}) - \lambda^{(k)}\mathbf{I}$ *on the left hand sides, that is, steps 1, 4, and 8 of Algorithm 8, where in this case* $\mathbf{A}_k$ *and* $\mathbf{C}_k$ *are diagonal matrices and* $\mathbf{B}_k$ *are tridiagonal. The numerical experiments show that solving the bordered system in Examples 2.3.2, 2.3.3 and 2.3.4 using LU factorization of* $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ *and BEM plus one iterative refinement give identical results at the root.*

		LU of $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$		BEM+1	
Example	Size of $\mathbf{A}(\gamma)$	Time/s	k	Time/s	k
2.3.2	961×961	0.32	7	0.09	8
2.3.3	961×961	0.32	9	0.09	10
2.3.4	3969×3969	12.02	7	1.01	7

Table 2.4: Comparing the cpu time between the solution obtained by direct LU factorization of $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ with one iterative refinement and BEM+1. The solver for $\mathbf{A}(\gamma^{(k)}) - \lambda^{(k)}\mathbf{I}$ in BEM is the Thomas algorithm for solving block tridiagonal systems. Clearly, BEM solver outperforms its LU factorization counterpart with the same number of iterative refinement.

From Table 2.4, Time/s represents the total time (in seconds) taken to solve for $f(\lambda, \gamma)$, $f_\lambda(\lambda, \gamma)$, $f_{\lambda\lambda}(\lambda, \gamma)$, $f_\gamma(\lambda, \gamma), f_{\lambda\gamma}(\lambda, \gamma)$ *and the corresponding* $\mathbf{x}$*'s. k represents the total number of iterations it took for Algorithm 3 to satisfy the desired tolerance as reported in the tables in Section 2.3. We remark that as we approach the root in the course of implementing BEM,* $\mathbf{A}(\gamma^{(k)}) - \lambda^{(k)}\mathbf{I}$ *became singular as predicted by the theory. Nevertheless, we did not encounter any problem with the program. This is because, only blockwise solves of the entries in* $\mathbf{A}(\gamma^{(k)}) - \lambda^{(k)}\mathbf{I}$ *are carried out in the Thomas algorithm. Besides, each* $\mathbf{\Gamma}_k$ *in* (2.43) *is not singular. It should be noted that without iterative refinement, the accuracy of the results obtained by BEM is of order* 10^{-12} *and* 10^{-13}*. A similar situation was encountered in using LU factorization on* $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$*. As a result of this, one step of iterative refinement, significantly improved the accuracy of the computed solution by BEM and LU factorization on Examples 2.3.2, 2.3.3, 2.3.4.*

Observe from the second row, third and fourth columns of Table 2.4 that while it took 0.32 seconds and 7 iterations for the computed solution obtained by LU fac-

torization of $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ *with one iterative refinement to converge to the desired tolerance, columns five and six show that BEM+1 takes 0.09 seconds and 9 iterations. In the same vein, from the third row, it took 0.32 seconds for the direct LU solver with* $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ *plus one iterative refinement and 0.09 seconds with BEM+1. It took 9 and 10 iterations respectively for both methods to converge to the desired tolerance.*

In the last row of Table 2.4, we see that when the size of the mesh is doubled, the total time taken to solve the five linear systems in Algorithm 3 involving $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ *on the right hand sides by LU factorization plus one iterative refinement is twelve times more than the time it took BEM+1. It should be mentioned that when systems involving* $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ *on the left hand sides were solved in Examples 2.3.2 and 2.3.3 using BEM with an LU solver, it took more time than with an LU solver on* $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$*. This is because BEM required more than one iterative refinements to attain the desired accuracy achieved by using LU on* $\mathbf{M}(\lambda^{(k)}, \gamma^{(k)})$ *in both examples. This is as a result of the near-singularity or singularity of* $\mathbf{A}(\gamma^{(k)}) - \lambda^{(k)}\mathbf{I}$*.*

In summary, as shown in Table 2.4, the goal of this section has been achieved as we have been able to solve the five bordered linear systems in Algorithm 3 more efficiently using BEM plus one iterative refinement.

In the next section, we present the special case of the theory developed in Section 2.2 in which λ is complex.

2.5 Implicit Determinant Method for a Parameter-Dependent Matrix when λ is Complex

Up till now, we have assumed that λ^* is real. In this section, we extend the theory to the case when λ^* is complex, and give an example.

When λ^* is complex in (2.4), we split λ into its real and imaginary parts as $\lambda = \alpha + i\beta$, where α and β are real. Hence, (2.11) can be rewritten as

$$\mathbf{G}(\mathbf{y}) = \begin{bmatrix} f(\alpha, \beta, \gamma) \\ f_\alpha(\alpha, \beta, \gamma) \\ f_\beta(\alpha, \beta, \gamma) \end{bmatrix} = \mathbf{0}, \tag{2.44}$$

where $\mathbf{y} = [\alpha, \beta, \gamma]^T$, and $f(\alpha, \beta, \gamma)$, $f_\alpha(\alpha, \beta, \gamma)$, $f_\beta(\alpha, \beta, \gamma)$ are complex. This means $\mathbf{G}(\mathbf{y}) = \mathbf{0}$, is six real nonlinear equations in three real unknowns, that

is,

$$\mathbf{G}(\mathbf{y}) = \begin{bmatrix} \mathrm{Re}f(\alpha,\beta,\gamma) \\ \mathrm{Re}f_\alpha(\alpha,\beta,\gamma) \\ \mathrm{Re}f_\beta(\alpha,\beta,\gamma) \\ \mathrm{Im}f(\alpha,\beta,\gamma) \\ \mathrm{Im}f_\alpha(\alpha,\beta,\gamma) \\ \mathrm{Im}f_\beta(\alpha,\beta,\gamma) \end{bmatrix} = \mathbf{0}.$$

However, in the course of the discussion in this section, this will reduce to a system of four real nonlinear equations in three real unknowns. This is due to the fact that $f_\beta(\alpha,\beta,\gamma) = if_\alpha(\alpha,\beta,\gamma)$ for all values of α,β and γ as we prove later. Hence, the last equation in (2.44) contains no extra information, and so it will be neglected. The resulting system of nonlinear equations will then be solved using the Gauss-Newton method.

This section is structured as follows, we begin by applying the theory of the implicit determinant method to

$$\mathbf{N}(\alpha,\beta,\gamma) = \mathbf{A}(\gamma) - \lambda\mathbf{I} = \mathbf{A}(\gamma) - (\alpha + i\beta)\mathbf{I},$$

where $\dim\mathcal{N}\big(\mathbf{N}(\alpha^*,\beta^*,\gamma^*)\big) = 1$, and $\mathbf{N}(\alpha^*,\beta^*,\gamma^*)$ has a 2-dimensional Jordan block corresponding to a zero eigenvalue. This will then be followed by presenting the four real nonlinear equations in three real unknowns. In Subsection 2.5.1, we show that the resulting Jacobian is of full rank and use the Gauss-Newton method in solving the over-determined nonlinear system of equations. The key results in this section are Lemma 2.5.1, Theorems 2.5.1, 2.5.2, 2.5.3, and Algorithm 11 is given for computing α^*, β^* and γ^*.

Write $\mathbf{M}$ from (2.2) in the form

$$\mathbf{M}(\alpha,\beta,\gamma) = \begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix},$$

where $\mathbf{N}(\alpha,\beta,\gamma) = \big(\mathbf{A} - (\alpha + i\beta)\mathbf{I}\big)$, $\mathbf{b}$ and $\mathbf{c}$ satisfy the conditions of Lemma 2.2.1 and consider the following system of linear equations

$$\begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(\alpha,\beta,\gamma) \\ f(\alpha,\beta,\gamma) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \tag{2.45}$$

Here, $\mathbf{N}(\alpha,\beta,\gamma)$ is complex and $\mathbf{b}$, $\mathbf{c}$ may be complex, so $\mathbf{x}(\alpha,\beta,\gamma)$ and $f(\alpha,\beta,\gamma)$ are complex. Observe that

$$\mathbf{N}_\alpha(\alpha,\beta,\gamma) = -\mathbf{I}, \quad \mathbf{N}_\beta(\alpha,\beta,\gamma) = -i\mathbf{I}, \quad \text{and} \quad \mathbf{N}_\gamma(\alpha,\beta,\gamma) = \mathbf{A}'(\gamma), \tag{2.46}$$

so $\mathbf{N}_\beta(\alpha,\beta,\gamma) = i\mathbf{N}_\alpha(\alpha,\beta,\gamma)$, where $\mathbf{N}_\alpha(\alpha,\beta,\gamma) = \frac{\mathrm{d}}{\mathrm{d}\alpha}\mathbf{N}(\alpha,\beta,\gamma)$ *e.t.c.*

Lemma 2.5.1. *If* $\mathbf{b}$, $\mathbf{c}$ *are chosen such that* $\mathbf{M}(\alpha^*,\beta^*,\gamma^*)$ *is nonsingular, then*

$$f_\beta(\alpha,\beta,\gamma) = if_\alpha(\alpha,\beta,\gamma),$$

for all α,β *and* γ.

Proof: After differentiating (2.45) with respect to α and using (2.46), we obtain

$$\begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\alpha(\alpha,\beta,\gamma) \\ f_\alpha(\alpha,\beta,\gamma) \end{bmatrix} = \begin{bmatrix} \mathbf{x}(\alpha,\beta,\gamma) \\ 0 \end{bmatrix}. \tag{2.47}$$

Again, if we differentiate both sides of (2.45) with respect to β, then with the help of (2.46),

$$\begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\beta(\alpha,\beta,\gamma) \\ f_\beta(\alpha,\beta,\gamma) \end{bmatrix} = \begin{bmatrix} i\mathbf{x}(\alpha,\beta,\gamma) \\ 0 \end{bmatrix}. \tag{2.48}$$

Note that both equations (2.47) and (2.48) have the same left hand side $\mathbf{M}(\alpha,\beta,\gamma)$ but with different right hand sides, though, the right hand side of (2.47) is a multiple of the right hand side of (2.48), that is,

$$\begin{bmatrix} \mathbf{x}_\beta(\alpha,\beta,\gamma) \\ f_\beta(\alpha,\beta,\gamma) \end{bmatrix} = i \begin{bmatrix} \mathbf{x}_\alpha(\alpha,\beta,\gamma) \\ f_\alpha(\alpha,\beta,\gamma) \end{bmatrix}, \tag{2.49}$$

for all α,β and γ. Hence, $f_\beta(\alpha,\beta,\gamma) = if_\alpha(\alpha,\beta,\gamma)$ and $\mathbf{x}_\beta(\alpha,\beta,\gamma) = i\mathbf{x}_\alpha(\alpha,\beta,\gamma)$ for all α,β and γ. ∎

The following fundamental result is a corollary of Theorem 2.2.1.

Theorem 2.5.1. *Let* $\mathbf{A}(\gamma^*)$ *be a real n by n matrix such that*

$$\dim \mathcal{N}\big(\mathbf{N}(\alpha^*,\beta^*,\gamma^*)\big) = 1.$$

Let zero be an eigenvalue of $\mathbf{N}(\alpha^*, \beta^*, \gamma^*)$ *corresponding to a 2-dimensional Jordan block. If* $\mathbf{b}$, $\mathbf{c}$ *are chosen such that* $\mathbf{M}(\alpha^*, \beta^*, \gamma^*)$ *is nonsingular, then*

1. $f(\alpha^*, \beta^*, \gamma^*) = 0$,
2. $f_\alpha(\alpha^*, \beta^*, \gamma^*) = 0$,
3. $f_\beta(\alpha^*, \beta^*, \gamma^*) = 0$,
4. $f_{\alpha\alpha}(\alpha^*, \beta^*, \gamma^*) \neq 0$.

Proof: From (2.45) and because $\dim \mathcal{N}\big(\mathbf{N}(\alpha^*, \beta^*, \gamma^*)\big) = 1$, the first part of the theorem follows from Lemma 2.2.2. By expanding the first n rows of (2.47) and evaluating at the root, one easily obtains

$$\mathbf{N}(\alpha^*, \beta^*, \gamma^*)\mathbf{x}_\alpha(\alpha^*, \beta^*, \gamma^*) + f_\alpha(\alpha^*, \beta^*, \gamma^*)\mathbf{b} = \mathbf{x}(\alpha^*, \beta^*, \gamma^*). \tag{2.50}$$

After premultiplying both sides by ${\psi^*}^H$, for all $\psi^* \in \mathcal{N}[\mathbf{N}(\alpha^*, \beta^*, \gamma^*)^H]\backslash\{\mathbf{0}\}$, we obtain

$${\psi^*}^H\mathbf{N}(\alpha^*, \beta^*, \gamma^*)\mathbf{x}_\alpha(\alpha^*, \beta^*, \gamma^*) + f_\alpha(\alpha^*, \beta^*, \gamma^*){\psi^*}^H\mathbf{b} = {\psi^*}^H\mathbf{x}(\alpha^*, \beta^*, \gamma^*).$$

The definition of ψ^* ensures that the first term on the left hand side equals zero, so that

$$f_\alpha(\alpha^*, \beta^*, \gamma^*) = \frac{{\psi^*}^H\mathbf{x}(\alpha^*, \beta^*, \gamma^*)}{{\psi^*}^H\mathbf{b}} = \frac{\tau{\psi^*}^H\phi^*}{{\psi^*}^H\mathbf{b}}.$$

Note that from Lemma 2.2.1, $\mathbf{x}(\alpha^*, \beta^*, \gamma^*) = \tau\phi^*$. Hence, $f_\alpha(\alpha^*, \beta^*, \gamma^*) = 0$ because ${\psi^*}^H\phi^* = 0$, (cf., Theorem 2.2.1). Thus,

$$\mathbf{N}(\alpha^*, \beta^*, \gamma^*)\mathbf{x}_\alpha(\alpha^*, \beta^*, \gamma^*) = \mathbf{x}(\alpha^*, \beta^*, \gamma^*),$$

is obvious from (2.50). This means $\mathbf{x}_\alpha(\alpha^*, \beta^*, \gamma^*)$ can be taken as the generalised eigenvector of $\mathbf{N}(\alpha^*, \beta^*, \gamma^*)$ corresponding to a double zero eigenvalue. Hence,

$${\psi^*}^H\mathbf{x}_\alpha(\alpha^*, \beta^*, \gamma^*) \neq 0, \tag{2.51}$$

by the dimensionality of the Jordan block.

Since (2.49) holds for all α, β and γ, we deduce that,

$$f_\beta(\alpha^*, \beta^*, \gamma^*) = i f_\alpha(\alpha^*, \beta^*, \gamma^*) = 0,$$

because $f_\alpha(\alpha^*, \beta^*, \gamma^*)$. Thus, proving the third part of the theorem. Next, if we differentiate both sides of (2.47) with respect to α and after some simplifications

$$\begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\alpha\alpha}(\alpha,\beta,\gamma) \\ f_{\alpha\alpha}(\alpha,\beta,\gamma) \end{bmatrix} = \begin{bmatrix} 2\mathbf{x}_\alpha(\alpha,\beta,\gamma) \\ 0 \end{bmatrix}. \tag{2.52}$$

This means

$$\mathbf{N}(\alpha,\beta,\gamma)\mathbf{x}_{\alpha\alpha}(\alpha,\beta,\gamma) + f_{\alpha\alpha}(\alpha,\beta,\gamma)\mathbf{b} = 2\mathbf{x}_\alpha(\alpha,\beta,\gamma),$$

and

$$f_{\alpha\alpha}(\alpha^*, \beta^*, \gamma^*) = \frac{2\boldsymbol{\psi}^{*H}\mathbf{x}_\alpha(\alpha^*, \beta^*, \gamma^*)}{\boldsymbol{\psi}^{*H}\mathbf{b}}. \tag{2.53}$$

Accordingly, $f_{\alpha\alpha}(\alpha^*, \beta^*, \gamma^*)$ is nonzero using (2.51).

■

Now, by differentiating both sides of (2.48) with respect to β,

$$\begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\beta\beta}(\alpha,\beta,\gamma) \\ f_{\beta\beta}(\alpha,\beta,\gamma) \end{bmatrix} = \begin{bmatrix} 2i\mathbf{x}_\beta(\alpha,\beta,\gamma) \\ 0 \end{bmatrix} = \begin{bmatrix} -2\mathbf{x}_\alpha(\alpha,\beta,\gamma) \\ 0 \end{bmatrix}. \tag{2.54}$$

Observe from (2.52) and (2.54) that for all α, β and γ,

$$\begin{bmatrix} \mathbf{x}_{\beta\beta}(\alpha,\beta,\gamma) \\ f_{\beta\beta}(\alpha,\beta,\gamma) \end{bmatrix} = -\begin{bmatrix} \mathbf{x}_{\alpha\alpha}(\alpha,\beta,\gamma) \\ f_{\alpha\alpha}(\alpha,\beta,\gamma) \end{bmatrix}. \tag{2.55}$$

The next result tells us more about the conditions satisfied by the partial derivatives of $f(\alpha, \beta, \gamma)$. The second part of the theorem makes use of (2.49).

Theorem 2.5.2. *Assume the conditions of Theorem 2.5.1 hold. If*

$$\boldsymbol{\psi}^{*H}\mathbf{A}'(\gamma^*)\mathbf{x}(\alpha^*, \beta^*, \gamma^*) \neq 0,$$

then $f_\gamma(\alpha^*, \beta^*, \gamma^*) \neq 0$. *Moreover,* $f_{\alpha\beta}(\alpha^*, \beta^*, \gamma^*) \neq 0$.

Proof: If we differentiate both sides of (2.45) with respect to γ, then

$$\begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\gamma(\alpha,\beta,\gamma) \\ f_\gamma(\alpha,\beta,\gamma) \end{bmatrix} = \begin{bmatrix} -\mathbf{A}'(\gamma)\mathbf{x}(\alpha,\beta,\gamma) \\ 0 \end{bmatrix}. \tag{2.56}$$

But by assumption, $\psi^{*H}\mathbf{A}'(\gamma^*)\mathbf{x}(\alpha^*,\beta^*,\gamma^*)$ is nonzero, this implies

$$f_\gamma(\alpha^*,\beta^*,\gamma^*) = -\frac{\psi^{*H}\mathbf{A}'(\gamma^*)\mathbf{x}(\alpha^*,\beta^*,\gamma^*)}{\psi^{*H}\mathbf{b}},$$

is also nonzero. Furthermore, by differentiating both sides of (2.47) with respect to β, and using (2.49) yields

$$\begin{aligned} \begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\alpha\beta}(\alpha,\beta,\gamma) \\ f_{\alpha\beta}(\alpha,\beta,\gamma) \end{bmatrix} &= \begin{bmatrix} \mathbf{x}_\beta(\alpha,\beta,\gamma) + i\mathbf{x}_\alpha(\alpha,\beta,\gamma) \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} 2i\mathbf{x}_\alpha(\alpha,\beta,\gamma) \\ 0 \end{bmatrix}. \end{aligned} \tag{2.57}$$

It is not difficult to see from (2.49) that

$$f_{\alpha\beta}(\alpha^*,\beta^*,\gamma^*) = \frac{2i\psi^{*H}\mathbf{x}_\alpha(\alpha^*,\beta^*,\gamma^*)}{\psi^{*H}\mathbf{b}}, \tag{2.58}$$

which is nonzero, since $\psi^{*H}\mathbf{x}_\alpha(\alpha^*,\beta^*,\gamma^*) \neq 0$. ∎

Notice that by virtue of the right hand sides of (2.52) and (2.57),

$$\begin{bmatrix} \mathbf{x}_{\alpha\beta}(\alpha,\beta,\gamma) \\ f_{\alpha\beta}(\alpha,\beta,\gamma) \end{bmatrix} = i \begin{bmatrix} \mathbf{x}_{\alpha\alpha}(\alpha,\beta,\gamma) \\ f_{\alpha\alpha}(\alpha,\beta,\gamma) \end{bmatrix}. \tag{2.59}$$

Observe that after differentiating both sides of (2.47) and (2.48) respectively with respect to γ, one easily obtains

$$\begin{bmatrix} \mathbf{N}(\alpha,\beta,\gamma) & \mathbf{b} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\alpha\gamma}(\alpha,\beta,\gamma) \\ f_{\alpha\gamma}(\alpha,\beta,\gamma) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_\gamma(\alpha,\beta,\gamma) - \mathbf{A}'(\gamma)\mathbf{x}_\alpha(\alpha,\beta,\gamma) \\ 0 \end{bmatrix}. \tag{2.60}$$

Since $f_\beta(\alpha,\beta,\gamma) = if_\alpha(\alpha,\beta,\gamma)$ for all α,β and γ from (2.49), it then means $f_\beta(\alpha,\beta,\gamma)$ contains no extra information. Consequently, we neglect the third

equation in (2.44), so that

$$\mathbf{G}(\mathbf{y}) = \begin{bmatrix} f(\alpha,\beta,\gamma) \\ f_\alpha(\alpha,\beta,\gamma) \end{bmatrix} = \mathbf{0}.$$

Therefore, the six over-determined nonlinear system in three real unknowns presented earlier reduces to the following four real over-determined nonlinear system of equations in three real unknowns,

$$\mathbf{F}(\mathbf{y}) = \begin{bmatrix} \mathrm{Re}[f(\alpha,\beta,\gamma)] \\ \mathrm{Re}[f_\alpha(\alpha,\beta,\gamma)] \\ \mathrm{Im}[f(\alpha,\beta,\gamma)] \\ \mathrm{Im}[f_\alpha(\alpha,\beta,\gamma)] \end{bmatrix} = \mathbf{0}, \tag{2.61}$$

where $\mathbf{F} : \mathbb{R}^3 \to \mathbb{R}^4$, $\mathbf{F}(\mathbf{y}) \in \mathbb{R}^4$ and $\mathbf{y} = [\alpha,\beta,\gamma]^T$. The Jacobian of $\mathbf{F}(\mathbf{y})$, can be expressed as

$$\mathbf{F}_\mathbf{y}(\mathbf{y}) = \begin{bmatrix} \mathrm{Re}[f_\alpha(\alpha,\beta,\gamma)] & \mathrm{Re}[f_\beta(\alpha,\beta,\gamma)] & \mathrm{Re}[f_\gamma(\alpha,\beta,\gamma)] \\ \mathrm{Re}f_{\alpha\alpha}(\alpha,\beta,\gamma) & \mathrm{Re}[f_{\alpha\beta}(\alpha,\beta,\gamma)] & \mathrm{Re}[f_{\alpha\gamma}(\alpha,\beta,\gamma)] \\ \mathrm{Im}[f_\alpha(\alpha,\beta,\gamma)] & \mathrm{Im}[f_\beta(\alpha,\beta,\gamma)] & \mathrm{Im}[f_\gamma(\alpha,\beta,\gamma)] \\ \mathrm{Im}[f_{\alpha\alpha}(\alpha,\beta,\gamma)] & \mathrm{Im}[f_{\alpha\beta}(\alpha,\beta,\gamma)] & \mathrm{Im}[f_{\alpha\gamma}(\alpha,\beta,\gamma)] \end{bmatrix} \in \mathbb{R}^{4\times 3}. \tag{2.62}$$

Next, we describe the Gauss-Newton method for solving the four real over-determined nonlinear system of equations for the three real unknown parameters, α, β and γ.

2.5.1 The Gauss-Newton Method for Solving (2.61)

In this section, we show that the rectangular Jacobian in (2.62) is of full rank at the root and apply the Gauss-Newton method to solve the four real over-determined nonlinear system of equations (2.61) in three real unknowns. The key result in this section is Theorem 2.5.3, and Algorithm 11 is given for computing the parameters α, β and γ.

From the algebra of complex numbers, since $f_\beta(\alpha,\beta,\gamma) = if_\alpha(\alpha,\beta,\gamma)$, this

is equivalent to $\mathrm{Re}[f_\beta(\alpha,\beta,\gamma)] = -\mathrm{Im}[f_\alpha(\alpha,\beta,\gamma)]$ and

$$\mathrm{Im}[f_\beta(\alpha,\beta,\gamma)] = \mathrm{Re}[f_\alpha(\alpha,\beta,\gamma)].$$

In the same vein, $\mathrm{Re}[f_{\alpha\beta}(\alpha,\beta,\gamma)] = -\mathrm{Im}[f_{\alpha\alpha}(\alpha,\beta,\gamma)]$ and

$$\mathrm{Im}[f_{\alpha\beta}(\alpha,\beta,\gamma)] = \mathrm{Re}[f_{\alpha\alpha}(\alpha,\beta,\gamma)],$$

because

$$f_{\alpha\beta}(\alpha,\beta,\gamma) = i f_{\alpha\alpha}(\alpha,\beta,\gamma).$$

Now, observe from (2.58), that $f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*) \neq 0$. This implies that either $\mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)]$ is nonzero and $\mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)$ is zero, or, $\mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)]$ is zero and $\mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)]$ is nonzero, or, both are nonzero. The same argument holds for the real and imaginary parts of $f_{\alpha\beta}(\alpha^*,\beta^*,\gamma^*)$. Hence, at the root, the Jacobian simplifies to

$$\mathbf{F}_{\mathbf{y}}(\mathbf{y}^*) = \begin{bmatrix} 0 & 0 & \mathrm{Re}[f_\gamma(\alpha^*,\beta^*,\gamma^*)] \\ \mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & -\mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & \mathrm{Im}[f_{\alpha\gamma}(\alpha^*,\beta^*,\gamma^*)] \\ 0 & 0 & \mathrm{Im}[f_\gamma(\alpha^*,\beta^*,\gamma^*)] \\ \mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & \mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & \mathrm{Im}[f_{\alpha\gamma}(\alpha^*,\beta^*,\gamma^*)] \end{bmatrix}. \tag{2.63}$$

The next theorem shows that the Jacobian above is of full rank.

Theorem 2.5.3. *Under the assumptions of Theorem 2.5.1 and if $f_\gamma(\alpha^*,\beta^*,\gamma^*) \neq 0$, then the Jacobian* (2.63) *is of full rank.*

Proof: If we can show that the unknowns p, q and r are zero in

$$\begin{bmatrix} 0 & 0 & \mathrm{Re}[f_\gamma(\alpha^*,\beta^*,\gamma^*)] \\ \mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & -\mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & \mathrm{Re}[f_{\alpha\gamma}(\alpha^*,\beta^*,\gamma^*)] \\ 0 & 0 & \mathrm{Im}[f_\gamma(\alpha^*,\beta^*,\gamma^*)] \\ \mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & \mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & \mathrm{Im}[f_{\alpha\gamma}(\alpha^*,\beta^*,\gamma^*)] \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \mathbf{0},$$

then the Jacobian is of full rank. From the first and third rows, and using the fact that $f_\gamma(\alpha^*,\beta^*,\gamma^*) \neq 0$, so that, at least one of $\mathrm{Re}f_\gamma(\alpha^*,\beta^*,\gamma^*)$ and $\mathrm{Im}f_\gamma(\alpha^*,\beta^*,\gamma^*)$ is nonzero. This implies, $r = 0$. This means that we are left

with

$$\mathbf{F}_{\mathbf{y}}(\mathbf{y}^*)_{[2\times 2]}\begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} \mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & -\mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] \\ \mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] & \mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)] \end{bmatrix}\begin{bmatrix} p \\ q \end{bmatrix} = \mathbf{0}.$$

The determinant of the matrix on the left hand side above is

$$\det[\mathbf{F}_{\mathbf{y}}(\mathbf{y}^*)_{[2\times 2]}] = (\mathrm{Re}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)])^2 + (\mathrm{Im}[f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)])^2.$$

But from Theorem 2.5.1, $f_{\alpha\alpha}(\alpha^*,\beta^*,\gamma^*)$ is nonzero. Accordingly, the determinant of $[\mathbf{F}_{\mathbf{y}}(\mathbf{y}^*)_{[2\times 2]}]$ is nonzero and $\mathbf{F}_{\mathbf{y}}(\mathbf{y}^*)_{[2\times 2]}$ is nonsingular . Therefore, $p = q = 0$ and so $\mathbf{F}_{\mathbf{y}}(\mathbf{y}^*)$ is of full rank. ∎

So far, because $f_\beta(\alpha,\beta,\gamma) = if_\alpha(\alpha,\beta,\gamma)$, we have been able to reduce the six real over-determined nonlinear system of equations in three real unknowns to four with the same number of unknowns. In addition, we have been able to show that the resulting Jacobian is of full rank at the root. Next, we describe an application of the Gauss-Newton method for the solution of the four real over-determined nonlinear system of equations in three real unknowns.

By an application of Gauss-Newton method (see, for example, [16, pp. 221-223]) to the nonlinear least squares problem

$$\min_{\mathbf{y}\in\mathbb{R}^3} \|\mathbf{F}(\mathbf{y})\|,$$

we solve (at least in theory)

$$[\mathbf{F}_{\mathbf{y}}(\mathbf{y}^{(k)})^T\mathbf{F}_{\mathbf{y}}(\mathbf{y}^{(k)})]\Delta\mathbf{y}^{(k)} = -\mathbf{F}_{\mathbf{y}}(\mathbf{y}^{(k)})^T\mathbf{F}(\mathbf{y}^{(k)}), \tag{2.64}$$

for $\Delta\mathbf{y}^{(k)}$ and update $\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)} + \Delta\mathbf{y}^{(k)}$. Computationally, we find the reduced QR factorization of $\mathbf{F}_{\mathbf{y}}(\mathbf{y}^{(k)}) = \mathbf{QR}$, where $\mathbf{Q} \in \mathbb{R}^{4\times 3}$ and $\mathbf{R} \in \mathbb{R}^{3\times 3}$. After making the necessary substitutions in (2.64), we obtain

$$\mathbf{R}\Delta\mathbf{y}^{(k)} = -\mathbf{Q}^T\mathbf{F}(\mathbf{y}^{(k)}).$$

Thus, the four real over-determined nonlinear equations in three real unknowns reduce to solving a square linear system of 3 equations for the 3 unknowns

$\Delta \mathbf{y}^{(k)} = [\Delta\alpha^{(k)}, \Delta\beta^{(k)}, \Delta\gamma^{(k)}]$ and updating

$$\begin{bmatrix} \alpha^{(k+1)} \\ \beta^{(k+1)} \\ \gamma^{(k+1)} \end{bmatrix} = \begin{bmatrix} \alpha^{(k)} \\ \beta^{(k)} \\ \gamma^{(k)} \end{bmatrix} + \begin{bmatrix} \Delta\alpha^{(k)} \\ \Delta\beta^{(k)} \\ \Delta\gamma^{(k)} \end{bmatrix}, \quad \text{for} \quad k = 0, 1, 2, \ldots.$$

Algorithm 11 Implicit Determinant on $\mathbf{N}(\alpha, \beta, \gamma)$ to find $[\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)}]^T$

Input: Choose $\alpha^{(0)}$, $\beta^{(0)}$, $\gamma^{(0)}$ and $\mathbf{b}, \mathbf{c} \in \mathbb{R}^n \backslash \{\mathbf{0}\}$ such that $\mathbf{M}(\alpha^{(0)}, \beta^{(0)}, \gamma^{(0)})$ is nonsingular, *tol*.

1: **for** $k = 0, 1, 2, \ldots,$ until convergence **do**
2: Solve (2.45) for $f(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$.
3: Use the $\mathbf{x}(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$ from the first step to compute $f_\alpha(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$ from (2.47).
4: Equate $f_\beta(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)}) = if_\alpha(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$.
5: With $\mathbf{x}_\alpha(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$ from step 2, solve (2.52) for $f_{\alpha\alpha}(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$.
6: Equate $f_{\alpha\beta}(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)}) = if_{\alpha\alpha}(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$.
7: Solve (2.56) for $f_\gamma(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$.
8: Solve (2.60) for $f_{\alpha\gamma}(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$.
9: Form $\mathbf{F}(\mathbf{y}^{(k)})$ from (2.61).
10: Find the reduced QR factorization of $\mathbf{F}_\mathbf{y}(\mathbf{y}^{(k)})$ as in (2.62).
11: Solve the 3 by 3 linear system

$$\mathbf{R}\Delta\mathbf{y}^{(k)} = -\mathbf{Q}^T\mathbf{F}(\mathbf{y}^{(k)}),$$

for $\Delta\mathbf{y}^{(k)}$.
12: Apply Newton update

$$\begin{bmatrix} \alpha^{(k+1)} \\ \beta^{(k+1)} \\ \gamma^{(k+1)} \end{bmatrix} = \begin{bmatrix} \alpha^{(k)} \\ \beta^{(k)} \\ \gamma^{(k)} \end{bmatrix} + \begin{bmatrix} \Delta\alpha^{(k)} \\ \Delta\beta^{(k)} \\ \Delta\gamma^{(k)} \end{bmatrix}.$$

13: **end for**

Output: $\mathbf{y}^{k_{\max}} = [\alpha^{(k_{\max})}, \beta^{(k_{\max})}, \gamma^{(k_{\max})}]$.

Algorithm 11 involves five linear solves with $\mathbf{M}(\alpha^{(k)}, \beta^{(k)}, \gamma^{(k)})$ on the left hand side but with different right hand sides. Hence, for each iteration, only one LU-factorization is computed while the L and U factors are used in the

five solves. The stopping condition for Algorithm 11 is

$$\|\Delta \mathbf{y}^{(k)}\| \leq tol. \tag{2.65}$$

Next, we give the following numerical example which illustrates the above theory.

Example 2.5.1. *Consider the following real parameter-dependent matrix*

$$A(\gamma) = \begin{bmatrix} -1 & 1 & 2 & 1 \\ \gamma & -1 & 0 & 2 \\ -2 & -1 & -1 & 1 \\ 0 & -2 & \gamma & -1 \end{bmatrix},$$

where γ represents power dispatch. This matrix was used in [19, p. 342] to illustrate exact resonance of two complex conjugate pairs of eigenvalues, which occurs when two damped oscillatory modes coalesce. In this example, we seek the value of γ^* such that $\mathbf{A}(\gamma^*)$ has a 2-dimensional Jordan block corresponding to the eigenvalue $\lambda^* = \alpha^* + i\beta^*$. In fact, $\gamma^* = 0$. The initial guesses for $\alpha^{(0)}$, $\beta^{(0)}$ and $\gamma^{(0)}$ are 2, 5 and 2 respectively. We chose $\mathbf{c}$ and $\mathbf{b}$ as $\mathbf{c} = [1,0,0,0]^T$ and $\mathbf{b} = \mathbf{A}'(\gamma)\mathbf{c}$. The computed values of α, β and γ are as tabulated in Table 2.5. From Table 2.5, we see that as we

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\gamma^{(k)}$	$\lvert\lambda^{(k+1)} - \lambda^{(k)}\rvert$	$\lvert\beta^{(k+1)} - \beta^{(k)}\rvert$	$\lvert\gamma^{(k+1)} - \gamma^{(k)}\rvert$	$\|\mathbf{G}(\mathbf{y}^{(k)})\|$
0	2.0	5.0	2e+00	3.5e+00	1.6e+00	1.2e+01	1.2e+01
1	-1.4	3.3	9e+00	4.3e-01	2.4e+00	1.4e+01	1.5e+01
2	-1.0	0.9	4e+00	1.9e+00	2.0e+00	5.2e+00	5.9e+00
3	-2.9	2.9	5e-01	1.1e+00	1.1e+00	1.8e+00	2.4e+00
4	-1.8	1.9	1e+00	8.8e-01	1.6e-01	1.8e+00	2.0e+00
5	-1.0	2.0	5e-01	2.3e-02	5.2e-02	5.2e-01	5.2e-01
6	-0.9	2.0	1e-02	4.9e-03	1.3e-02	1.4e-02	2.0e-02
7	-1.0	2.0	3e-07	3.1e-05	2.2e-05	3.9e-07	3.8e-05
8	-1.0	2.0	9e-10	2.6e-12	4.2e-13	9.1e-10	9.1e-10
9	-1.0	2.0	5e-20	0.0e+00	0.0e+00	5.1e-20	5.2e-20

Table 2.5: Values of $\alpha^{(k)}$, $\beta^{(k)}$ and $\gamma^{(k)}$ obtained after applying Algorithm 11 on $\mathbf{A}(\gamma)$. From the last row, it will be observed that $\alpha^* = -1$, $\beta^* = 2$ and $\gamma^* = 0$. Column 8 show that the results converged quadratically as predicted by Theorem 2.5.2.

approach the root (in this context, 'root' is the same as resonance), we observe quadratic convergence in column eight. This agrees with the standard Gauss-Newton theory.

Besides, at $\gamma^* = 0$, our computations show that $\alpha^* = -1$, $\beta^* = 2$, and $\lambda^* = -1 + 2i$, (as in [19]). Moreover, the computed values of $f_\gamma(\alpha^*, \beta^*, \gamma^*)$ and $f_{\alpha\alpha}(\lambda^*, \beta^*, \gamma^*)$ are -2 and 2-2i respectively, which are nonzero as required by Theorems 2.5.1 (part 4) and 2.5.2.

2.6 Conclusion

The aim of this chapter has been achieved, in the sense that given a real parameter-dependent nonsymmetric matrix $\mathbf{A}(\gamma)$, we have been able to extend the implicit determinant method of Spence and Poulton in [55] to obtain numerical algorithms for determining when two eigenvalues of $\mathbf{A}(\gamma)$ move together and coalesce as the parameter γ is varied thereby forming a 2-dimensional Jordan block. The algorithms are based on Newton's method and we provide conditions under which they achieve quadratic convergence. Results of numerical experiments are given which confirm the theory. Be that as it may, the algorithms relies on close enough initial guesses to the desired values of λ and γ.

CHAPTER 3

The Calculation of the Distance to a Nearby Defective Matrix

3.1 Introduction

Let $\mathbf{A}$ be a complex n by n matrix with n distinct eigenvalues. It is a classic problem in numerical linear algebra to find

$$\mathrm{d}(\mathbf{A}) = \inf\{\|\mathbf{A} - \mathbf{B}\| : \quad \mathbf{B} \quad \text{is a defective matrix}\},$$

where $\|\cdot\| = \|\cdot\|_F$ or $\|\cdot\| = \|\cdot\|_2$. Hence, $\mathrm{d}(\mathbf{A})$ is the distance of the matrix $\mathbf{A}$ to the set of matrices which have a Jordan block of at least dimension 2. In this chapter, given a simple matrix $\mathbf{A}$, we present two numerical methods to find a nearby defective matrix from $\mathbf{A}$ and the distance between them. For the first method, we extend the implicit determinant method used in Chapter 2 to formulate the problem as a real system of three nonlinear equations in three real unknowns which will be solved by Newton's method.

The second method is more straightforward but less elegant. Assuming the nearest defective matrix is real, we simply write down all the equations involving all the unknowns, and obtain a real system of $(2n + 3)$ nonlinear equations in $(2n + 2)$ real unknowns (we do not consider the complex case here). Though not guaranteed to find the nearest defective matrix, since Newton's method provides no such guarantees, in all the examples considered our

methods did, in fact, find the nearest defective matrix and hence $\mathrm{d}(\mathbf{A})$ was computed.

The distance of a simple matrix to a defective matrix is linked with the sensitivity analysis of eigenvalues. The condition number of a simple eigenvalue λ is given by $1/|\mathbf{y}^H\mathbf{x}|$, (see [62]) where $\mathbf{x}$ and $\mathbf{y}$ are normalised right and left eigenvectors respectively corresponding to λ. For a defective eigenvalue, we have $\mathbf{y}^H\mathbf{x} = 0$. Therefore, the condition number of the defective eigenvalue λ, is infinite.

However, it is well-known that even if the eigenvalues of a matrix are simple and well-separated from each other, they can be ill-conditioned [62]. Hence the measure of the distance $\mathrm{d}(\mathbf{A})$ of a matrix $\mathbf{A}$ to a defective matrix $\mathbf{B}$ is important for determining the sensitivity of an eigendecomposition. There is a very informative discussion and history of this problem in [5], where the contributions of Demmel [13; 14] and Wilkinson [65; 66] are discussed in detail. Another important paper is that by Lippert and Edelman [36], who use ideas from differential geometry and singularity theory to discuss the sensitivity of double eigenvalues. In particular, they present a condition that measures the ill-conditioning of a matrix with a 2-dimensional Jordan block. Section 1.2 of Chapter one, contains some more literature reviews on this topic. The key paper that provides the solution to the nearest defective matrix problem is that of Alam & Bora [4] who provide both the theory and an algorithm based on pseudospectra.

Following Trefethen and Embree [61], the ε-pseudospectrum $\Lambda_\varepsilon(\mathbf{A})$ of a matrix $\mathbf{A}$ is given by

$$\Lambda_\varepsilon(\mathbf{A}) = \{\sigma_{\min}(\mathbf{A} - z\mathbf{I}) < \varepsilon\},$$

where $\varepsilon > 0$, $\sigma_{\min}$ denotes the smallest singular value and $z \in \mathbb{C}$. Equivalently,

$$\Lambda_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} \,|\, \det(\mathbf{A} + \mathbf{E} - z\mathbf{I}) = 0, \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\| < \varepsilon\}.$$

If $\Lambda_\varepsilon(\mathbf{A})$ has n components, then $\mathbf{A} + \mathbf{E}$ has n distinct eigenvalues for all perturbation matrices $\mathbf{E} \in \mathbb{C}^{n\times n}$ with $\|\mathbf{E}\| < \varepsilon$ and hence, $\mathbf{A} + \mathbf{E}$ is not defective. Alam and Bora [4] take these ideas and seek the smallest perturbation matrix $\mathbf{E}$ such that the pseudospectra of $\mathbf{A} + \mathbf{E}$ coalesce. They present the following

theorem (see [4, Theorem 4.1] and [5, Lemma 1]).

Theorem 3.1.1. *Let* $\mathbf{A} \in \mathbb{C}^{n\times n}$ *and* $z \in \mathbb{C} \setminus \Lambda(\mathbf{A})$, *so that* $\mathbf{A} - z\mathbf{I}$ *has a simple smallest singular value* $\varepsilon > 0$ *with corresponding left and right singular vectors* $\mathbf{u}$ *and* $\mathbf{v}$ *such that* $(\mathbf{A} - z\mathbf{I})\mathbf{v} = \varepsilon\mathbf{u}$. *Then* z *is an eigenvalue of* $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ *with geometric multiplicity* 1 *and corresponding left and right eigenvectors* $\mathbf{u}$ *and* $\mathbf{v}$ *respectively. Furthermore, if* $\mathbf{u}^H\mathbf{v} = 0$, *then* z *has algebraic multiplicity two which is greater than one (its geometric multiplicity), hence it is a defective eigenvalue of* $\mathbf{B}$ *and* $\|\mathbf{A} - \mathbf{B}\| = \varepsilon$.

Proof: First, we want to show that z is an eigenvalue of $\mathbf{B}$. To do this, we subtract $z\mathbf{I}$ from both sides of $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ to obtain

$$\mathbf{B} - z\mathbf{I} = \mathbf{A} - z\mathbf{I} - \varepsilon\mathbf{u}\mathbf{v}^H,$$

and by post multiplying both sides by the right singular vector $\mathbf{v}$ we have,

$$(\mathbf{B} - z\mathbf{I})\mathbf{v} = (\mathbf{A} - z\mathbf{I} - \varepsilon\mathbf{u}\mathbf{v}^H)\mathbf{v} = (\mathbf{A} - z\mathbf{I})\mathbf{v} - \varepsilon\mathbf{u} = \mathbf{0},$$

by assumption. Hence, $\mathbf{B}\mathbf{v} = z\mathbf{v}$. In a similar fashion, it can be shown that $\mathbf{u}^H\mathbf{B} = z\mathbf{u}^H$. This shows that $\mathbf{u}$ and $\mathbf{v}$ are left and right eigenvectors corresponding to the eigenvalue z. Next, we want to show that the geometric multiplicity of z is one. From $(\mathbf{B} - z\mathbf{I})\mathbf{v} = (\mathbf{A} - z\mathbf{I} - \varepsilon\mathbf{u}\mathbf{v}^H)\mathbf{v}$, and using the singular value decomposition

$$\mathbf{A} - z\mathbf{I} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H = \sum_{k=1}^{n} \sigma_k \mathbf{u}_k \mathbf{v}_k^H,$$

where $\mathbf{u} = \mathbf{u}_n$, $\sigma_n = \varepsilon$ and $\mathbf{v} = \mathbf{v}_n$. Thus, since $\sigma_n = \varepsilon$ is a simple singular value of $(\mathbf{B} - z\mathbf{I})$,

$$\begin{aligned}
\mathbf{B} - z\mathbf{I} &= \sum_{k=1}^{n} \sigma_k \mathbf{u}_k \mathbf{v}_k^H - \varepsilon\mathbf{u}\mathbf{v}^H \\
&= \sum_{k=1}^{n-1} \sigma_k \mathbf{u}_k \mathbf{v}_k^H + \sigma_n \mathbf{u}_n \mathbf{v}_n^H - \varepsilon\mathbf{u}_n\mathbf{v}_n^H \\
&= \sum_{k=1}^{n-1} \sigma_k \mathbf{u}_k \mathbf{v}_k^H \\
&= \mathbf{U}_{n-1}\mathbf{\Sigma}_{n-1}\mathbf{V}_{n-1}^H.
\end{aligned}$$

This shows that the rank of $\mathbf{B} - z\mathbf{I}$ is $n-1$. Hence z has a geometric multiplicity of 1[1]. Furthermore, if $\mathbf{u}^H\mathbf{v} = 0$, then standard Jordan theory shows that there exists a generalised eigenvector $\hat{\mathbf{v}}$ corresponding to the eigenvalue z such that $(\mathbf{B} - z\mathbf{I})\hat{\mathbf{v}} = \mathbf{v}$. Hence, $(\mathbf{B} - z\mathbf{I})^2\hat{\mathbf{v}} = \mathbf{0}$ with $\hat{\mathbf{v}} \neq \mathbf{0}$. Therefore, the algebraic multiplicity of z is greater than one (its geometric multiplicity), hence z is a defective eigenvalue of $\mathbf{B}$.

■

Theorem 3.1.1 leads to the result $\mathbf{E} := -\varepsilon\mathbf{u}\mathbf{v}^H$ so that $\mathbf{B} = \mathbf{A} + \mathbf{E}$ is a defective matrix and

$$\mathrm{d}(\mathbf{A}) = \|\mathbf{E}\| = \varepsilon\|\mathbf{u}\mathbf{v}^H\| = \varepsilon,$$

since $\mathbf{v}^H\mathbf{v} = \mathbf{u}^H\mathbf{u} = 1$. One drawback of the algorithm in [4] is that it is rather expensive since it involves repeated calculation of pseudospectra. Also a decision on when two pseudospectral curves coalesce is required. In [5], a method based on calculating lowest generalised saddle points of singular values is described. This has the advantage that it is able to deal with the nongeneric case when $\mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ is ill-conditioned. We shall present a straightforward, yet elegant and very fast method that deals with the generic case when $\mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ is well-conditioned.

Using the notation of Theorem 3.1.1 the problem is to find $z \in \mathbb{C}$, $\mathbf{u}, \mathbf{v} \in \mathbb{C}^n$ and $\varepsilon \in \mathbb{R}$ such that

$$(\mathbf{A} - z\mathbf{I})\mathbf{v} - \varepsilon\mathbf{u} = \mathbf{0}, \tag{3.1}$$

$$\varepsilon\mathbf{v} - (\mathbf{A} - z\mathbf{I})^H\mathbf{u} = \mathbf{0}, \tag{3.2}$$

and

$$\mathbf{u}^H\mathbf{v} = 0. \tag{3.3}$$

Following Theorem 3.1.1 and Lippert and Edelman [36, Sections 4 and 5] we make the following assumption.

Assumption 3.1.1. *Assume* $\mathbf{A} - z\mathbf{I}$ *satisfies the conditions of Theorem 3.1.1 and that* $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ *is well-conditioned. That is, with* $z = \alpha + i\beta$, *the* 2×2 *matrix*

[1] $\mathrm{rank}(\mathbf{B} - z\mathbf{I}) = n - \dim\mathcal{N}(\mathbf{B} - z\mathbf{I}) = n - 1$. This means that the dimension of the nullspace of $\mathbf{B} - z\mathbf{I}$ is 1 or $\mathbf{B}$ has a geometric multiplicity of 1.

$\begin{bmatrix} \varepsilon_{\alpha\alpha} & \varepsilon_{\alpha\beta} \\ \varepsilon_{\alpha\beta} & \varepsilon_{\beta\beta} \end{bmatrix}$ *is well-conditioned, where* $\varepsilon_{\alpha\alpha}$ *denotes the second partial derivative of* ε *with respect to* α*, etc. (see [36, Theorem 5.1 and Corollary 5.2]).*

This chapter is organised as follows. Section 3.2 contains some background theory and the derivation of the implicit determinant method to solve the nearest defective matrix problem. Section 3.3 describes Newton's method applied to this problem and in Section 3.4 we give numerical examples that illustrate the power of our approach. An alternative approach based on solving (3.1), (3.2) and (3.3) with normalisations of $\mathbf{u}$ and $\mathbf{v}$ is presented in Section 3.5. This is not as elegant as the one in Sections 3.2-3.4, yet gives the same results.

3.2 The Implicit Determinant Method to find a Nearby Defective Matrix

In this section, we describe some background theory and present our numerical approach to finding a nearby defective matrix from a simple one, which is formulated as solving a real 3-dimensional nonlinear system. We emphasise that, since our numerical method uses standard Newton's method to solve the nonlinear system, we cannot guarantee that it finds the nearest defective matrix. Therefore, the use of the word 'nearby'. However, a more sophisticated nonlinear solver may be used if greater reliability were sought. We do not do this here because in all our examples the nearest defective matrix was found using standard Newton's method.

First, we formulate the problem following Alam and Bora [4, Section 4]. Equations (3.1)-(3.2) can be written as

$$\begin{bmatrix} -\varepsilon\mathbf{I} & \mathbf{A} - z\mathbf{I} \\ (\mathbf{A} - z\mathbf{I})^H & -\varepsilon\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \mathbf{0}. \tag{3.4}$$

Set $z = \alpha + i\beta$, $\mathbf{x} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}$ and consider the Hermitian matrix [4]

$$\mathbf{K}(\alpha, \beta, \varepsilon) = \begin{bmatrix} -\varepsilon\mathbf{I} & \mathbf{A} - (\alpha + i\beta)\mathbf{I} \\ \big(\mathbf{A} - (\alpha + i\beta)\mathbf{I}\big)^H & -\varepsilon\mathbf{I} \end{bmatrix}. \tag{3.5}$$

Clearly, by the Hermitian property of $\mathbf{K}(\alpha,\beta,\varepsilon)$, $\mathbf{x}$ is both a right and left nullvector of $\mathbf{K}(\alpha,\beta,\varepsilon)$. Now, it is not difficult to see that,

$$\mathbf{K}_\alpha(\alpha,\beta,\varepsilon) = \begin{bmatrix} & -\mathbf{I} \\ -\mathbf{I} & \end{bmatrix}; \quad \text{and} \quad \mathbf{K}_\beta(\alpha,\beta,\varepsilon) = i\begin{bmatrix} & -\mathbf{I} \\ \mathbf{I} & \end{bmatrix};$$

with $\mathbf{K}_\varepsilon(\alpha,\beta,\varepsilon) = -\mathbf{I}_{2n}$. From above, it is obvious that $\mathbf{K}_{\alpha\varepsilon}(\alpha,\beta,\varepsilon) = \mathbf{O}$, $\mathbf{K}_{\alpha\alpha}(\alpha,\beta,\varepsilon) = \mathbf{O}$, $\mathbf{K}_{\beta\beta}(\alpha,\beta,\varepsilon) = \mathbf{O}$, *e.t.c.* The following Lemma follows immediately from Assumption 3.1.1.

Lemma 3.2.1. *Let $\varepsilon > 0$ satisfy the conditions in Theorem 3.1.1. Furthermore, let $z = \alpha + i\beta$ be such that $\mathbf{K}(\alpha,\beta,\varepsilon)\mathbf{x} = 0$, where $\mathbf{x} = \begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix} \in \mathbb{C}^{2n}\backslash\{\mathbf{0}\}$. Then $dim\,\mathcal{N}\big(\mathbf{K}(\alpha,\beta,\varepsilon)\big) = 1$.*

Proof: If ε is a simple singular value of $(\mathbf{A} - z\mathbf{I})$, then the right and left singular vectors $\mathbf{v}$ and $\mathbf{u}$ in (3.1) and (3.2) are uniquely defined (up to their sign). Hence, there exists only one vector $\mathbf{x} = \begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix}$ (up to sign) which satisfies $\mathbf{K}(\alpha,\beta,\varepsilon)\mathbf{x} = \mathbf{0}$ and hence the result follows. ∎

We now introduce an algorithm to find the critical values of α, β and ε such that the Hermitian matrix $\mathbf{K}(\alpha,\beta,\varepsilon)$ is singular at the root and the constraint on $\mathbf{x}$ given by (3.3) is satisfied. We use the implicit determinant method, introduced in [55] to find photonic band structure in periodic materials such as photonic crystals. In [22], the implicit determinant method was used to find a 2-dimensional Jordan block in a Hamiltonian matrix in order to calculate the distance to instability and in Chapter 2 it was used to compute a 2-dimensional Jordan block in a parameter-dependent nonsymmetric matrix. Here, we have a three-parameter problem with a constraint to satisfy.

First, we introduce the bordered matrix $\mathbf{M}(\alpha,\beta,\gamma)$, defined in (3.6) below. The next theorem gives conditions to ensure that this matrix is nonsingular.

Theorem 3.2.1. *Let $(\alpha^*,\beta^*,\varepsilon^*,\mathbf{x}^*)$ solve*

$$\mathbf{K}(\alpha,\beta,\varepsilon)\mathbf{x}(\alpha,\beta,\varepsilon) = \mathbf{0}, \quad \mathbf{x}(\alpha,\beta,\varepsilon) \neq \mathbf{0},$$

so that $\dim \mathcal{N}\big(\mathbf{K}(\alpha^*, \beta^*, \varepsilon^*)\big) = 1$ *and* $\mathbf{x}^* \in \mathcal{N}\big(\mathbf{K}(\alpha^*, \beta^*, \varepsilon^*)\big) \setminus \{\mathbf{0}\}$. *For some* $\mathbf{c} \in \mathbb{C}^{2n} \setminus \{\mathbf{0}\}$, *assume*

$$\mathbf{c}^H \mathbf{x}^* \neq 0.$$

Then the Hermitian matrix

$$\mathbf{M}(\alpha, \beta, \varepsilon) = \begin{bmatrix} \mathbf{K}(\alpha, \beta, \varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix}, \tag{3.6}$$

is nonsingular at $\alpha = \alpha^*$, $\beta = \beta^*$, $\varepsilon = \varepsilon^*$.

Proof: This result follows from the proof of the first part of Lemma 2.2.1. ■

Now consider the following linear system

$$\begin{bmatrix} \mathbf{K}(\alpha, \beta, \varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ f \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}, \tag{3.7}$$

where $\mathbf{K}(\alpha, \beta, \varepsilon)$ is given by (3.5). As $\mathbf{M}(\alpha^*, \beta^*, \varepsilon^*)$ is nonsingular we have that $\mathbf{M}(\alpha, \beta, \varepsilon)$ is nonsingular for α, β and ε in the vicinity of α^*, β^* and ε^*. Theorem 3.2.1 implies that both $\mathbf{x}$ and f are smooth functions of α, β and ε near $(\alpha^*, \beta^*, \varepsilon^*)$, and so we write $\mathbf{x} = \mathbf{x}(\alpha, \beta, \gamma)$, $f = f(\alpha, \beta, \gamma)$ and (3.7) as

$$\begin{bmatrix} \mathbf{K}(\alpha, \beta, \varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(\alpha, \beta, \varepsilon) \\ f(\alpha, \beta, \varepsilon) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \tag{3.8}$$

Applying Cramer's rule to (3.8), we obtain

$$f(\alpha, \beta, \varepsilon) = \frac{\det \mathbf{K}(\alpha, \beta, \varepsilon)}{\det \mathbf{M}(\alpha, \beta, \varepsilon)}.$$

Since $\mathbf{M}(\alpha, \beta, \varepsilon)$ is nonsingular in the neighbourhood of $(\alpha^*, \beta^*, \varepsilon^*)$, then by Theorem 3.2.1 there is an equivalence between the zero eigenvalue of $\mathbf{K}(\alpha, \beta, \varepsilon)$ (which we are looking for) and the zeros of $f(\alpha, \beta, \varepsilon)$. Hence, to find the values of α, β and ε such that $\det \mathbf{K}(\alpha, \beta, \varepsilon) = 0$ we seek the solutions of

$$f(\alpha, \beta, \varepsilon) = 0. \tag{3.9}$$

If $f(\alpha^*, \beta^*, \varepsilon^*) = 0$, then the first row of (3.8) reduces to

$$\mathbf{K}(\alpha^*, \beta^*, \varepsilon^*)\mathbf{x}(\alpha^*, \beta^*, \varepsilon^*) = \mathbf{0}, \tag{3.10}$$

that is, $\mathbf{x}(\alpha^*, \beta^*, \varepsilon^*) = \mathbf{x}^*$ is an eigenvector of $\mathbf{K}(\alpha^*, \beta^*, \varepsilon^*)$ belonging to the eigenvalue zero. For the following derivation we use the notation

$$\mathbf{x}(\alpha, \beta, \varepsilon) = \begin{bmatrix} \mathbf{u}(\alpha, \beta, \varepsilon) \\ \mathbf{v}(\alpha, \beta, \varepsilon) \end{bmatrix}. \tag{3.11}$$

Note also that since $\mathbf{K}(\alpha, \beta, \varepsilon)$ and $\mathbf{M}(\alpha, \beta, \varepsilon)$ are Hermitian, $f(\alpha, \beta, \varepsilon)$ is real. Differentiating both sides of (3.8) with respect to α leads to

$$\begin{bmatrix} \mathbf{K}(\alpha, \beta, \varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\alpha(\alpha, \beta, \varepsilon) \\ f_\alpha(\alpha, \beta, \varepsilon) \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_\alpha(\alpha, \beta, \varepsilon)\mathbf{x}(\alpha, \beta, \varepsilon) \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{v}(\alpha, \beta, \varepsilon) \\ \mathbf{u}(\alpha, \beta, \varepsilon) \\ 0 \end{bmatrix}. \tag{3.12}$$

Expanding along the first row gives

$$\mathbf{K}(\alpha, \beta, \varepsilon)\mathbf{x}_\alpha(\alpha, \beta, \varepsilon) + \mathbf{c} f_\alpha(\alpha, \beta, \varepsilon) = \begin{bmatrix} \mathbf{v}(\alpha, \beta, \varepsilon) \\ \mathbf{u}(\alpha, \beta, \varepsilon) \end{bmatrix}. \tag{3.13}$$

Multiplying this equation, evaluated at $(\alpha^*, \beta^*, \varepsilon^*)$, from the left by the eigenvector $\mathbf{x}^{*H}$ of $\mathbf{K}(\alpha^*, \beta^*, \varepsilon^*)$ gives

$$\begin{aligned} f_\alpha(\alpha^*, \beta^*, \varepsilon^*) &= \begin{bmatrix} \mathbf{u}^{*H} & \mathbf{v}^{*H} \end{bmatrix} \begin{bmatrix} \mathbf{v}^* \\ \mathbf{u}^* \end{bmatrix} \\ &= \mathbf{u}^{*H}\mathbf{v}^* + \mathbf{v}^{*H}\mathbf{u}^* \\ &= 2\mathrm{Re}(\mathbf{u}^{*H}\mathbf{v}^*), \end{aligned}$$

where we have used $\mathbf{x}^{*H}\mathbf{c} = 1$ from (3.8). Similarly, differentiating both sides

of (3.8) with respect to β, gives

$$\begin{bmatrix} \mathbf{K}(\alpha,\beta,\varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_\beta(\alpha,\beta,\varepsilon) \\ f_\beta(\alpha,\beta,\varepsilon) \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_\beta(\alpha,\beta,\varepsilon)\mathbf{x}(\alpha,\beta,\varepsilon) \\ 0 \end{bmatrix} = i \begin{bmatrix} \mathbf{v}(\alpha,\beta,\varepsilon) \\ -\mathbf{u}(\alpha,\beta,\varepsilon) \\ 0 \end{bmatrix}. \tag{3.14}$$

Again, evaluating at $(\alpha^*,\beta^*,\varepsilon^*)$ and multiplying by ${\mathbf{x}^*}^H$ from the left leads to

$$\begin{aligned} f_\beta(\alpha^*,\beta^*,\varepsilon^*) &= i \begin{bmatrix} {\mathbf{u}^*}^H & {\mathbf{v}^*}^H \end{bmatrix} \begin{bmatrix} \mathbf{v}^* \\ -\mathbf{u}^* \end{bmatrix} \\ &= i({\mathbf{u}^*}^H\mathbf{v}^* - {\mathbf{v}^*}^H\mathbf{u}^*) \\ &= -2\mathrm{Im}({\mathbf{u}^*}^H\mathbf{v}^*). \end{aligned}$$

Clearly,

$$f_\alpha(\alpha^*,\beta^*,\varepsilon^*) = 0 \quad \text{and} \quad f_\beta(\alpha^*,\beta^*,\varepsilon^*) = 0 \quad \Longleftrightarrow \quad {\mathbf{u}^*}^H\mathbf{v}^* = 0.$$

Thus, we have reduced the problem of finding a solution to

$$\det \mathbf{K}(\alpha^*,\beta^*,\varepsilon^*) = 0,$$

with ${\mathbf{u}^*}^H\mathbf{v}^* = 0$, to that of solving $\mathbf{g}(\alpha,\beta,\varepsilon) = \mathbf{0}$, where

$$\mathbf{g}(\alpha,\beta,\varepsilon) = \begin{bmatrix} f(\alpha,\beta,\varepsilon) \\ f_\alpha(\alpha,\beta,\varepsilon) \\ f_\beta(\alpha,\beta,\varepsilon) \end{bmatrix}, \tag{3.15}$$

which is a real system of three nonlinear equations in three real unknowns. In the next section we describe the solution procedure using Newton's method.

3.3 Newton's method applied to $\mathbf{g}(\alpha,\beta,\varepsilon) = \mathbf{0}$

In this section, we describe how to implement Newton's method for the nonlinear system $\mathbf{g}(\alpha,\beta,\varepsilon) = \mathbf{0}$. We also obtain a nondegeneracy condition that ensures nonsingularity of the Jacobian matrix of $\mathbf{g}(\alpha,\beta,\varepsilon)$ at the root, and hence

confirms that Newton's method converges quadratically for a close enough starting guess. The nondegeneracy condition is shown to be equivalent to one introduced by Lippert and Edelman [36] for the conditioning of the 2-dimensional Jordan block of $\mathbf{B} = \mathbf{A} - \varepsilon \mathbf{u}\mathbf{v}^H$ (see Assumption 3.1.1). The main result in this section is Lemma 3.3.1 and Algorithm 12 is given for computing the values of the parameters α, β and ε.

Newton's method applied to $\mathbf{g}(\alpha, \beta, \varepsilon)$ is given by

$$\mathbf{G}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)}) \begin{bmatrix} \Delta\alpha^{(k)} \\ \Delta\beta^{(k)} \\ \Delta\varepsilon^{(k)} \end{bmatrix} = -\mathbf{g}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)}), \tag{3.16}$$

where $\alpha^{(k+1)} = \alpha^{(k)} + \Delta\alpha^{(k)}$, $\beta^{(k+1)} = \beta^{(k)} + \Delta\beta^{(k)}$ and $\varepsilon^{(k+1)} = \varepsilon^{(k)} + \Delta\varepsilon^{(k)}$, for $k = 0, 1, 2 \ldots$ until convergence, with a starting guess $(\alpha^{(0)}, \beta^{(0)}, \varepsilon^{(0)})$, and where the Jacobian is

$$\mathbf{G}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)}) = \begin{bmatrix} f_\alpha^{(k)} & f_\beta^{(k)} & f_\varepsilon^{(k)} \\ f_{\alpha\alpha}^{(k)} & f_{\alpha\beta}^{(k)} & f_{\alpha\varepsilon}^{(k)} \\ f_{\beta\alpha}^{(k)} & f_{\beta\beta}^{(k)} & f_{\beta\varepsilon}^{(k)} \end{bmatrix}, \tag{3.17}$$

and all the matrix entries are evaluated at $(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})$. The values of $f^{(k)}$, $f_\alpha^{(k)}$ and $f_\beta^{(k)}$ are found using (3.8), (3.12) and (3.14). For the remaining values, we differentiate (3.8), (3.12) and (3.14) with respect to ε, that is,

$$\begin{bmatrix} \mathbf{K}(\alpha, \beta, \varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} x_\varepsilon(\alpha, \beta, \varepsilon) \\ f_\varepsilon(\alpha, \beta, \varepsilon) \end{bmatrix} = \begin{bmatrix} -\mathbf{K}_\varepsilon(\alpha, \beta, \varepsilon) \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{x}(\alpha, \beta, \varepsilon) \\ 0 \end{bmatrix}, \tag{3.18}$$

and

$$\begin{bmatrix} \mathbf{K}(\alpha, \beta, \varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\alpha\varepsilon}(\alpha, \beta, \varepsilon) \\ f_{\alpha\varepsilon}(\alpha, \beta, \varepsilon) \end{bmatrix} = \begin{bmatrix} -[\mathbf{K}_\alpha(\alpha, \beta, \varepsilon)\mathbf{x}_\varepsilon(\alpha, \beta, \varepsilon) + \mathbf{K}_\varepsilon(\alpha, \beta, \varepsilon)\mathbf{x}_\alpha(\alpha, \beta, \varepsilon)] \\ 0 \end{bmatrix} \tag{3.19}$$

$$= \begin{bmatrix} \mathbf{v}_\varepsilon(\alpha, \beta, \varepsilon) + \mathbf{u}_\alpha(\alpha, \beta, \varepsilon) \\ \mathbf{u}_\varepsilon(\alpha, \beta, \varepsilon) + \mathbf{v}_\alpha(\alpha, \beta, \varepsilon) \\ 0 \end{bmatrix},$$

as well as

$$\begin{bmatrix} \mathbf{K}(\alpha,\beta,\varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\beta\varepsilon}(\alpha,\beta,\varepsilon) \\ f_{\beta\varepsilon}(\alpha,\beta,\varepsilon) \end{bmatrix} = \begin{bmatrix} -[\mathbf{K}_\beta(\alpha,\beta,\varepsilon)\mathbf{x}_\varepsilon(\alpha,\beta,\varepsilon) + \mathbf{K}_\varepsilon(\alpha,\beta,\varepsilon)\mathbf{x}_\beta(\alpha,\beta,\varepsilon)] \\ 0 \end{bmatrix} \tag{3.20}$$

$$= \begin{bmatrix} i\mathbf{v}_\varepsilon(\alpha,\beta,\varepsilon) + \mathbf{u}_\beta(\alpha,\beta,\varepsilon) \\ -i\mathbf{u}_\varepsilon(\alpha,\beta,\varepsilon) + \mathbf{v}_\beta(\alpha,\beta,\varepsilon) \\ 0 \end{bmatrix},$$

in order to find $f_\varepsilon^{(k)}$, $f_{\alpha\varepsilon}^{(k)}$ and $f_{\beta\varepsilon}^{(k)}$ respectively. Furthermore, by differentiating both sides of (3.12) with respect to α and β, we obtain

$$\begin{bmatrix} \mathbf{K}(\alpha,\beta,\varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\alpha\alpha}(\alpha,\beta,\varepsilon) \\ f_{\alpha\alpha}(\alpha,\beta,\varepsilon) \end{bmatrix} = \begin{bmatrix} -2\mathbf{K}_\alpha(\alpha,\beta,\varepsilon)\mathbf{x}_\alpha(\alpha,\beta,\varepsilon) \\ 0 \end{bmatrix} = 2 \begin{bmatrix} \mathbf{v}_\alpha(\alpha,\beta,\varepsilon) \\ \mathbf{u}_\alpha(\alpha,\beta,\varepsilon) \\ 0 \end{bmatrix}, \tag{3.21}$$

and

$$\begin{bmatrix} \mathbf{K}(\alpha,\beta,\varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\alpha\beta}(\alpha,\beta,\varepsilon) \\ f_{\alpha\beta}(\alpha,\beta,\varepsilon) \end{bmatrix} = \begin{bmatrix} -[\mathbf{K}_\beta(\alpha,\beta,\varepsilon)\mathbf{x}_\alpha(\alpha,\beta,\varepsilon) + \mathbf{K}_\alpha(\alpha,\beta,\varepsilon)\mathbf{x}_\beta(\alpha,\beta,\varepsilon)] \\ 0 \end{bmatrix} \tag{3.22}$$

$$= \begin{bmatrix} i\mathbf{v}_\alpha(\alpha,\beta,\varepsilon) + \mathbf{v}_\beta(\alpha,\beta,\varepsilon) \\ -i\mathbf{u}_\alpha(\alpha,\beta,\varepsilon) + \mathbf{u}_\beta(\alpha,\beta,\varepsilon) \\ 0 \end{bmatrix},$$

to compute $f_{\alpha\alpha}^{(k)}$ and $f_{\alpha\beta}^{(k)} = f_{\beta\alpha}^{(k)}$ respectively. Finally, differentiate both sides of (3.14) with respect to β to get

$$\begin{bmatrix} \mathbf{K}(\alpha,\beta,\varepsilon) & \mathbf{c} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_{\beta\beta}(\alpha,\beta,\varepsilon) \\ f_{\beta\beta}(\alpha,\beta,\varepsilon) \end{bmatrix} = \begin{bmatrix} -2\mathbf{K}_\beta(\alpha,\beta,\varepsilon)\mathbf{x}_\beta(\alpha,\beta,\varepsilon) \\ 0 \end{bmatrix} \tag{3.23}$$

$$= 2i \begin{bmatrix} \mathbf{v}_\beta(\alpha,\beta,\varepsilon) \\ -\mathbf{u}_\beta(\alpha,\beta,\varepsilon) \\ 0 \end{bmatrix}.$$

Therefore, in order to evaluate the components of $\mathbf{G}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})$ and $\mathbf{g}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})$ we only need to solve the linear systems above, which, importantly, all have the same Hermitian system matrix $\mathbf{M}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})$. Hence, only one LU factorisation of $\mathbf{M}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})$ per iteration in Newton's method is required. Note that Newton's method itself is only carried out in three dimensions. Next, we state the Newton-based algorithm for this problem.

Algorithm 12 Newton's method for computing α, β and ε.

Input: Given $(\alpha^{(0)}, \beta^{(0)}, \varepsilon^{(0)})$ and $\mathbf{c} \in \mathbb{C}^{2n} \backslash \{\mathbf{0}\}$ such that $\mathbf{M}(\alpha^{(0)}, \beta^{(0)}, \varepsilon^{(0)})$ is nonsingular; set $k = 0$:

1: Solve (3.8) and (3.12) and (3.14) in order to evaluate

$$\mathbf{g}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)}) = \begin{bmatrix} f(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)}) \\ f_\alpha(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)}) \\ f_\beta(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)}) \end{bmatrix}.$$

2: Solve (3.18), (3.21), (3.22), (3.23), (3.19) and (3.20) in order to evaluate the Jacobian $\mathbf{G}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})$ given by (3.17).

3: Newton update: Solve (3.16) in order to get $(\alpha^{(k+1)}, \beta^{(k+1)}, \varepsilon^{(k+1)})$

4: Repeat until convergence.

Output: $\alpha^*, \beta^*, \varepsilon^*$

Finally we show, that provided a certain nondegeneracy condition holds, the Jacobian $\mathbf{G}$ is nonsingular at the root. In the limit we have

$$\mathbf{G}(\alpha^*, \beta^*, \varepsilon^*) = \begin{bmatrix} 0 & 0 & f_\varepsilon^* \\ f_{\alpha\alpha}^* & f_{\alpha\beta}^* & f_{\alpha\varepsilon}^* \\ f_{\beta\alpha}^* & f_{\beta\beta}^* & f_{\beta\varepsilon}^* \end{bmatrix}, \tag{3.24}$$

since $f_\alpha^* = f_\beta^* = 0$. Multiplying the first row of (3.18) evaluated at $(\alpha^*, \beta^*, \varepsilon^*)$ from the left by ${\mathbf{x}^*}^H$ gives

$$f_\varepsilon(\alpha^*, \beta^*, \varepsilon^*) = {\mathbf{x}^*}^H \mathbf{x}^* \neq 0,$$

(recall ${\mathbf{x}^*}^H \mathbf{c} = 1$ from (3.8)). Hence, the Jacobian (3.24) is nonsingular if and only if

$$F_{\alpha\beta}^* := f_{\alpha\alpha}^* f_{\beta\beta}^* - {f_{\alpha\beta}^*}^2 \neq 0, \quad \text{since} \quad f_{\alpha\beta}^* = f_{\beta\alpha}^*. \tag{3.25}$$

With similar calculations as before we obtain

$$f_{\alpha\alpha}(\alpha^*,\beta^*,\varepsilon^*) = 2\mathbf{x}^{*H}\begin{bmatrix}\mathbf{v}_\alpha^*\\ \mathbf{u}_\alpha^*\end{bmatrix}, \quad f_{\beta\beta}(\alpha^*,\beta^*,\varepsilon^*) = 2i\mathbf{x}^{*H}\begin{bmatrix}\mathbf{v}_\beta^*\\ -\mathbf{u}_\beta^*\end{bmatrix}, \tag{3.26}$$

and

$$f_{\alpha\beta}(\alpha^*,\beta^*,\varepsilon^*) = \mathbf{x}^{*H}\left(i\begin{bmatrix}\mathbf{v}_\alpha^*\\ -\mathbf{u}_\alpha^*\end{bmatrix} + \begin{bmatrix}\mathbf{v}_\beta^*\\ \mathbf{u}_\beta^*\end{bmatrix}\right). \tag{3.27}$$

Lemma 3.3.1. : *Under Assumption 3.1.1,* $F_{\alpha\beta}^* = f_{\alpha\alpha}^* f_{\beta\beta}^* - {f_{\alpha\beta}^*}^2 \neq 0$.

Proof: If ε is a simple singular value of $(\mathbf{A} - (\alpha+\beta i)\mathbf{I})$, $\alpha,\beta \in \mathbb{R}$, so that

$$\big(\mathbf{A} - (\alpha+\beta i)\mathbf{I}\big)\mathbf{v} = \varepsilon\mathbf{u}, \quad \big(\mathbf{A} - (\alpha+\beta i)\mathbf{I}\big)^H\mathbf{u} = \varepsilon\mathbf{v},$$

then (see Sun [58]) ε, $\mathbf{u}$ and $\mathbf{v}$ are smooth functions of α and β. Furthermore, Lippert and Edelman [36, Theorem 3.1] show that if ${\mathbf{u}^*}^H\mathbf{v}^* = 0$, then $\varepsilon_\alpha^* := \varepsilon_\alpha(\alpha^*,\beta^*) = 0$, $\varepsilon_\beta^* := \varepsilon_\beta(\alpha^*,\beta^*) = 0$ and $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}^*{\mathbf{v}^*}^H$ has a 2-dimensional Jordan block. In addition, the ill-conditioning of the matrix $\mathbf{B}$ is determined by the ill-conditioning of

$$\mathbf{E} = \begin{bmatrix}\varepsilon_{\alpha\alpha}^* & \varepsilon_{\alpha\beta}^*\\ \varepsilon_{\alpha\beta}^* & \varepsilon_{\beta\beta}^*\end{bmatrix},$$

see [36, Corollary 5.2]. Under Assumption 3.1.1 we have $\det(\mathbf{E}) \neq 0$. Recall (3.4) and (3.5) with $\varepsilon = \varepsilon(\alpha,\beta)$, $\mathbf{v} = \mathbf{v}(\alpha,\beta)$, $\mathbf{u} = \mathbf{u}(\alpha,\beta)$ and $\mathbf{x} = \begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix}$. This means (3.4) and (3.5) can be rewritten as

$$\begin{bmatrix}-\varepsilon(\alpha,\beta)\mathbf{I} & \mathbf{A} - (\alpha+i\beta)\mathbf{I}\\ [\mathbf{A} - (\alpha+i\beta)]^H & -\varepsilon(\alpha,\beta)\mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{u}(\alpha,\beta)\\ \mathbf{v}(\alpha,\beta)\end{bmatrix} = \mathbf{0}. \tag{3.28}$$

Differentiate both sides with respect to α to obtain

$$\begin{aligned}&\begin{bmatrix}-\varepsilon_\alpha(\alpha,\beta)\mathbf{I} & -\mathbf{I}\\ -\mathbf{I} & -\varepsilon_\alpha(\alpha,\beta)\mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{u}(\alpha,\beta)\\ \mathbf{v}(\alpha,\beta)\end{bmatrix}\\ +&\begin{bmatrix}-\varepsilon(\alpha,\beta)\mathbf{I} & \mathbf{A} - (\alpha+i\beta)\mathbf{I}\\ [\mathbf{A} - (\alpha+i\beta)]^H & -\varepsilon(\alpha,\beta)\mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{u}_\alpha(\alpha,\beta)\\ \mathbf{v}_\alpha(\alpha,\beta)\end{bmatrix} = \mathbf{0}.\end{aligned} \tag{3.29}$$

Again, by differentiating both sides of the above with respect to α, yields

$$\begin{bmatrix} -\varepsilon_{\alpha\alpha}(\alpha,\beta)\mathbf{I} & \mathbf{O} \\ \mathbf{O} & -\varepsilon_{\alpha\alpha}(\alpha,\beta)\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{u}(\alpha,\beta) \\ \mathbf{v}(\alpha,\beta) \end{bmatrix} + 2 \begin{bmatrix} -\varepsilon_{\alpha}(\alpha,\beta)\mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & -\varepsilon_{\alpha}(\alpha,\beta)\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{u}_{\alpha}(\alpha,\beta) \\ \mathbf{v}_{\alpha}(\alpha,\beta) \end{bmatrix} + \begin{bmatrix} -\varepsilon(\alpha,\beta)\mathbf{I} & \mathbf{A}-(\alpha+i\beta)\mathbf{I} \\ [\mathbf{A}-(\alpha+i\beta)]^H & -\varepsilon(\alpha,\beta)\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{u}_{\alpha\alpha}(\alpha,\beta) \\ \mathbf{v}_{\alpha\alpha}(\alpha,\beta) \end{bmatrix} = \mathbf{0}.$$

Now, using the fact that at the root $\varepsilon_{\alpha}(\alpha,\beta) = 0$, then

$$-\varepsilon_{\alpha\alpha}(\alpha^*,\beta^*)\mathbf{x}(\alpha^*,\beta^*) - 2 \begin{bmatrix} \mathbf{v}_{\alpha}(\alpha^*,\beta^*) \\ \mathbf{u}_{\alpha}(\alpha^*,\beta^*) \end{bmatrix} + \mathbf{K}(\alpha^*,\beta^*,\varepsilon^*) \begin{bmatrix} \mathbf{u}_{\alpha\alpha}(\alpha^*,\beta^*) \\ \mathbf{v}_{\alpha\alpha}(\alpha^*,\beta^*) \end{bmatrix} = \mathbf{0}.$$

So that

$$\mathbf{K}(\alpha^*,\beta^*,\varepsilon^*)\mathbf{x}_{\alpha\alpha}(\alpha^*,\beta^*) - 2 \begin{bmatrix} \mathbf{v}_{\alpha}(\alpha^*,\beta^*) \\ \mathbf{u}_{\alpha}(\alpha^*,\beta^*) \end{bmatrix} = \mathbf{x}(\alpha^*,\beta^*)\varepsilon_{\alpha\alpha}(\alpha^*,\beta^*). \tag{3.30}$$

After premultiplying both sides by ${\mathbf{x}^*}^H = \mathbf{x}(\alpha^*,\beta^*)$, we have

$${\mathbf{x}^*}^H\mathbf{K}(\alpha^*,\beta^*,\varepsilon^*)\mathbf{x}_{\alpha\alpha}(\alpha^*,\beta^*) - 2{\mathbf{x}^*}^H \begin{bmatrix} \mathbf{v}_{\alpha}(\alpha^*,\beta^*) \\ \mathbf{u}_{\alpha}(\alpha^*,\beta^*) \end{bmatrix} = ({\mathbf{x}^*}^H\mathbf{x}^*)\varepsilon_{\alpha\alpha}(\alpha^*,\beta^*).$$

The first term on the left hand side is zero because $\mathbf{x}^*$ is both a left and right nullvector of $\mathbf{K}(\alpha^*,\beta^*,\varepsilon^*)$, so that the above reduces to

$$2{\mathbf{x}^*}^H \begin{bmatrix} \mathbf{v}_{\alpha}(\alpha^*,\beta^*) \\ \mathbf{u}_{\alpha}(\alpha^*,\beta^*) \end{bmatrix} = -({\mathbf{x}^*}^H\mathbf{x}^*)\varepsilon_{\alpha\alpha}(\alpha^*,\beta^*).$$

Using the definition for $f_{\alpha\alpha}(\alpha^*,\beta^*)$ from (3.26) in the above expression, then

$$f_{\alpha\alpha}(\alpha^*,\beta^*) = 2{\mathbf{x}^*}^H \begin{bmatrix} \mathbf{v}_{\alpha}(\alpha^*,\beta^*) \\ \mathbf{u}_{\alpha}(\alpha^*,\beta^*) \end{bmatrix} = -({\mathbf{x}^*}^H\mathbf{x}^*)\varepsilon_{\alpha\alpha}(\alpha^*,\beta^*).$$

Taking the second partial derivative of both sides of (3.29) with respect to β

and evaluating at the root using $\varepsilon_\alpha(\alpha^*, \beta^*) = \varepsilon_\beta(\alpha^*, \beta^*) = 0$ we obtain

$$\mathbf{K}(\alpha^*, \beta^*, \varepsilon^*)\mathbf{x}^*_{\alpha\beta} + \begin{bmatrix} -i\mathbf{v}^*_\alpha - \mathbf{v}^*_\beta \\ i\mathbf{u}^*_\alpha - \mathbf{u}^*_\beta \end{bmatrix} = \varepsilon^*_{\alpha\beta}\mathbf{x}^*. \tag{3.31}$$

Similarly, it can be shown by taking first and second partial derivatives of both sides of (3.28) with respect to β and evaluating at the root, that

$$\mathbf{K}(\alpha^*, \beta^*, \varepsilon^*)\mathbf{x}^*_{\beta\beta} + 2i \begin{bmatrix} -\mathbf{v}^*_\beta \\ \mathbf{u}^*_\beta \end{bmatrix} = \varepsilon^*_{\beta\beta}\mathbf{x}^*. \tag{3.32}$$

Premultiplying both sides of (3.31) and (3.32) by the eigenvector ${\mathbf{x}^*}^H$ of $\mathbf{K}(\alpha^*, \beta^*, \varepsilon^*)$, we obtain respectively

$$f^*_{\alpha\beta} = -({\mathbf{x}^*}^H\mathbf{x}^*)\varepsilon^*_{\alpha\beta} \quad \text{and} \quad f^*_{\beta\beta} = -({\mathbf{x}^*}^H\mathbf{x}^*)\varepsilon^*_{\beta\beta}, \tag{3.33}$$

where we have used (3.26) and (3.27). Therefore,

$$F^*_{\alpha\beta} = f^*_{\alpha\alpha}f^*_{\beta\beta} - {f^*_{\alpha\beta}}^2 = ({\mathbf{x}^*}^H\mathbf{x}^*)^2\left[\varepsilon^*_{\alpha\alpha}\varepsilon^*_{\beta\beta} - {\varepsilon^*_{\alpha\beta}}^2\right] = ({\mathbf{x}^*}^H\mathbf{x}^*)^2 \det(\mathbf{E}) \neq 0,$$

since $\det(\mathbf{E}) \neq 0$ and $\mathbf{x}^* \neq 0$. ∎

In summary, Lemma 3.3.1 shows that when the defective matrix $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ is well-conditioned, Algorithm 12 should exhibit quadratic convergence for a close enough starting guess.

Next, we present a brief discussion on how to choose optimum starting vectors for Algorithm 12.

3.3.1 Optimal Starting Vectors when A is Nonnormal

In this subsection, given a simple nonnormal matrix $\mathbf{A}$, we discuss a systematic way to choose good starting guesses for computing $\mathrm{d}(\mathbf{A})$ and a nearby defective matrix from $\mathbf{A}$.

1. Reduce matrix to Schur form (this is not necessary in cases where $\mathbf{A}$ is already in upper triangular form).

2. Take $z^{(0)}$ as the average of the two smallest diagonal elements, for example, for the Kahan matrix we could take $z^{(0)} = \frac{s^{n-1}+s^n}{2}$, or a value close to the average of two diagonal elements where the defective eigenvalues is suspected to be lurking (if they are known before hand), *e.g.*, Trefethen and Wilkinson matrix.

3. Find the singular value decomposition of $(\mathbf{A} - z^{(0)}\mathbf{I})$.

4. Choose $\varepsilon^{(0)}$ as the minimum singular value of $(\mathbf{A} - z^{(0)}\mathbf{I})$.

5. Choose $\mathbf{u}^{(0)}$, $\mathbf{v}^{(0)}$, as the left and right singular vectors respectively corresponding to the smallest singular value $\varepsilon^{(0)}$ of $(\mathbf{A} - z^{(0)}\mathbf{I})$, $\mathbf{x}^{(0)} = [\mathbf{u}^{(0)}, \mathbf{v}^{(0)}]^T$, so that $\mathbf{c} = \mathbf{x}^{(0)}$.

3.4 Numerical Experiments

We now illustrate the numerical performance of our method with several examples which are taken from [4]. As has been mentioned earlier, since our method is based on Newton's method it finds a nearby defective matrix. We cannot guarantee it finds the nearest defective matrix. However, in all cases considered here, our method found the nearest defective matrix according to Alam and Bora [4] (but at much less cost, of course). Throughout this section, $\Delta\mathbf{y}^{(k)} = [\Delta\alpha^{(k)}, \Delta\beta^{(k)}, \Delta\varepsilon^{(k)}]^T$.

Example 3.4.1. *Consider the matrix* $\mathbf{A} = \begin{bmatrix} -1 & 5 \\ 0 & -2 \end{bmatrix}$, *(see [61]). As initial guesses we choose* $\alpha^{(0)} = \beta^{(0)} = 0$, $\varepsilon^{(0)} = \sigma_{\min}$, $\mathbf{u}^{(0)} = \mathbf{u}_{\min}$ *and* $v^{(0)} = \mathbf{v}_{\min}$, *where* $\sigma_{\min}$ *is the minimum singular value of* $\mathbf{A}$ *with corresponding left and right singular vectors* $\mathbf{u}_{\min}$ *and* $\mathbf{v}_{\min}$. $\mathbf{x}^{(0)}$ *is determined from* (3.11) *and* $\mathbf{c} = \mathbf{x}^{(0)}$. *We stop the iteration once*

$$\|\mathbf{g}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})\| < \tau, \quad \textit{where} \quad \tau = 10 \times 10^{-15}.$$

Table 3.1 shows the results for Example 3.4.1. Hence, $z = -1.5$ *is a degenerate common boundary point of the pseudospectrum, according to [4]. With* $\varepsilon = 4.9510 \times 10^{-2}$, $\mathbf{u} = [-9.8538 \times 10^{-2}, \ -9.9513 \times 10^{-1}]$ *and* $\mathbf{v} = [9.9513 \times 10^{-1}, \ -9.8538 \times 10^{-2}]$ *we have that* $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ *is a defective matrix. The last*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)},\beta^{(k)},\varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	0.0000e+00	0	3.6597e-01	2.5725e-01	2.3e+00	2.7421e-02
1	-2.0400e+00	0	6.8361e-01	4.5671e-01	6.7e-01	-1.0604e-01
2	-1.6498e+00	0	1.4010e-01	6.5424e-02	1.7e-01	-5.4992e-02
3	-1.5063e+00	0	5.5504e-02	3.6573e-03	8.6e-03	-4.5647e-02
4	-1.5000e+00	0	4.9522e-02	7.8979e-06	2.0e-05	-4.5473e-02
5	-1.5000e+00	0	4.9510e-02	4.5572e-11	1.1e-10	-4.5473e-02
6	-1.5000e+00	0	4.9510e-02	1.6022e-17	7.5e-17	-4.5473e-02

Table 3.1: Columns five and six shows quadratic convergence for Example 3.4.1.

column of Table 3.1 shows the value of $F_{\alpha\beta}^{(k)} = f_{\alpha\alpha}^{(k)} f_{\beta\beta}^{(k)} - f_{\alpha\beta}^{(k)^2}$ *(given by* (3.25)*) and we see that the final value* $F_{\alpha\beta}^{*} \neq 0$ *at the root. Algorithm 12 converges quadratically in* 6 *iterations, as expected from Newton's method.*

Example 3.4.2. *Let* $\mathbf{A} \in \mathbb{C}^{n\times n}$ *be the Kahan matrix [61], which is given by*

$$\mathbf{A} = \begin{bmatrix} 1 & -c & -c & -c & -c \\ & s & -sc & -sc & -sc \\ & & s^2 & -s^2c & -s^2c \\ & & & \ddots & \vdots \\ & & & & s^{n-1} \end{bmatrix}, \tag{3.34}$$

where $s^{n-1} = 0.1$ *and* $s^2 + c^2 = 1$*. We consider this matrix for* $n = 6, 15, 20$*. The starting values and stopping condition are chosen as in Example 3.4.1.*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)},\beta^{(k)},\varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	0.0000e+00	0	9.9694e-03	8.1049e-02	1.4e-01	3.9318e-01
1	1.3643e-01	0	1.2145e-02	3.9165e-02	1.2e-02	-1.0032e+00
2	1.3319e-01	0	7.1339e-04	4.3976e-03	5.5e-03	-4.5529e-01
3	1.2767e-01	0	4.9351e-04	8.2870e-05	4.8e-05	-4.3191e-01
4	1.2763e-01	0	4.7049e-04	4.7344e-08	7.7e-08	-4.3136e-01
5	1.2763e-01	0	4.7049e-04	5.3585e-15	1.8e-14	-4.3136e-01
6	1.2763e-01	0	4.7049e-04	1.5099e-17	2.2e-17	-4.3136e-01

Table 3.2: Results for Example 3.4.2, $n = 6$. Quadratic convergence is seen in column five for $k = 4$ and 5.

Table 3.2 shows the results for $n = 6$*. In this case the two smallest eigenvalues of*

A *i.e.,* 1.5849×10^{-1} *and* 10^{-1} *coalesce at* 1.2763×10^{-1} *for a value of* $\varepsilon = 4.7049 \times 10^{-4}$. *It means* $z^* = 1.2763 \times 10^{-1}$ *is a double eigenvalue of the nearby defective matrix* $\mathbf{B} = \mathbf{A} - \varepsilon \mathbf{u}\mathbf{v}^H$ *with,* $\varepsilon^* = 4.7049 \times 10^{-4}$ *and the computed value of* $\mathbf{x}^* = [\mathbf{u}, \mathbf{v}]^T$, $\mathbf{u}$ *and* $\mathbf{v}$ *are the left and right eigenvectors of* $\mathbf{B}$ *corresponding to the eigenvalue* z^*. *The last column of Table 3.2 shows the value of* $F_{\alpha\beta}^{(k)}$, *which is not close to zero. The quadratic convergence rate is observed in rows five and six of column five.*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	0.0000e+00	0	4.7454e-04	6.1760e-03	5.9e-02	5.3943e-03
1	5.9261e-02	0	2.5746e-04	1.2796e-03	2.5e-02	-5.3968e-03
2	8.3781e-02	0	8.5281e-06	8.9265e-05	9.1e-03	6.1204e-01
3	9.2879e-02	0	4.5688e-09	2.2854e-04	1.0e-02	1.5419e-03
4	1.0301e-01	0	2.0519e-06	4.1072e-05	3.6e-03	-2.1635e-04
5	1.0659e-01	0	5.9562e-07	6.4229e-06	6.8e-04	-1.0397e-04
6	1.0727e-01	0	5.2063e-07	1.9724e-07	2.2e-05	-9.0471e-05
7	1.0729e-01	0	5.1757e-07	2.2164e-10	2.5e-08	-9.0078e-05
8	1.0729e-01	0	5.1757e-07	2.7618e-16	3.1e-14	-9.0077e-05
9	1.0729e-01	0	5.1757e-07	3.7721e-18	4.0e-16	-9.0077e-05

Table 3.3: In Example 3.4.2, for $n = 15$, superlinear convergence is observed for $k = 6, 7, 8$ and 9 in columns five and six.

Table 3.3 shows the results for $n = 15$. *In this case the two smallest eigenvalues of* **A** *i.e.,* 1.1788×10^{-1} *and* 10^{-1} *coalesce at* 1.0729×10^{-1} *for a value of* $\varepsilon = 5.1757 \times 10^{-7}$. *It means* $z^* = 1.0729 \times 10^{-1}$ *is a double eigenvalue of the nearby defective matrix* $\mathbf{B} = \mathbf{A} - \varepsilon \mathbf{u}\mathbf{v}^H$ *with,* $\varepsilon^* = 5.1757 \times 10^{-7}$ *and the computed value of* $\mathbf{x}^* = [\mathbf{u}, \mathbf{v}]^T$, $\mathbf{u}$ *and* $\mathbf{v}$ *are the left and right eigenvectors of* $\mathbf{B}$ *corresponding to the eigenvalue* z^*.

Table 3.4 shows the results for $n = 20$. *In this case the two smallest eigenvalues of* **A** *i.e.,* 1.1288×10^{-1} *and* 10^{-1} *coalesce at* 1.0501×10^{-1} *for a value of* $\varepsilon = 2.8841 \times 10^{-8}$. *This means that* $z^* = 1.0501 \times 10^{-1}$ *is a double eigenvalue of the nearby defective matrix* $\mathbf{B} = \mathbf{A} - \varepsilon \mathbf{u}\mathbf{v}^H$ *with,* $\varepsilon^* = 2.8841 \times 10^{-8}$ *and the computed value of* $\mathbf{x}^* = [\mathbf{u}, \mathbf{v}]^T$, $\mathbf{u}$ *and* $\mathbf{v}$ *are the left and right eigenvectors of* $\mathbf{B}$ *corresponding to the eigenvalue* z^*.

From the last columns in Tables 3.2-3.4, we see that the value of $F_{\alpha\beta}^{(k)}$ *becomes smaller as the size of the Kahan matrix becomes large. This means the matrix* $\mathbf{B} = \mathbf{A} - \varepsilon \mathbf{u}\mathbf{v}^H$ *becomes increasingly ill-conditioned as n increases. We also observed a corresponding deterioration in the rate of convergence of Newton's method as the value*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)},\beta^{(k)},\varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	0.0000e+00	0	1.3141e-04	2.0010e-03	4.7e-02	7.2389e-04
1	4.7216e-02	0	5.7554e-05	5.0411e-04	2.5e-02	-1.0374e-03
2	7.2398e-02	0	4.8678e-06	1.5028e-04	2.1e-02	-5.2893e-04
3	9.3454e-02	0	2.5086e-06	1.7045e-05	6.5e-03	-1.7880e-05
4	9.9991e-02	0	9.4991e-09	7.7825e-06	8.1e-03	-2.2361e-05
5	1.0812e-01	0	1.0316e-07	3.0809e-06	4.8e-03	-3.8308e-07
6	1.0332e-01	0	4.4010e-08	2.0224e-06	1.5e-03	-2.3190e-06
7	1.0482e-01	0	3.0835e-08	2.0280e-07	1.9e-04	-1.3074e-06
8	1.0501e-01	0	2.8867e-08	3.2047e-09	3.1e-06	-1.2248e-06
9	1.0501e-01	0	2.8841e-08	8.2531e-13	7.9e-10	-1.2236e-06
10	1.0501e-01	0	2.8841e-08	1.3443e-18	1.1e-15	-1.2236e-06

Table 3.4: Results for Example 3.4.2 for $n = 20$. The above table shows that $k = 7, 8, 9, 10$, we obtained superlinear convergence in columns 5 and 6.

of $F_{\alpha\beta}^{(k)}$ becomes smaller, which is consistent with the theory.

Example 3.4.3. *Let $\mathbf{A} \in \mathbb{C}^{n\times n}$ be the Frank matrix taken from the Matlab gallery $\mathbf{A} = gallery('frank', n)$, for $n = 6, 12$. As initial guesses we choose $\alpha^{(0)} = \beta^{(0)} = 0$, $\varepsilon^{(0)} = \sigma_{\min}$, $\mathbf{u}^{(0)} = \mathbf{u}_{\min}$ and $\mathbf{v}^{(0)} = \mathbf{v}_{\min}$, where $\sigma_{\min}$ is the minimum singular value of $\mathbf{A}$ with corresponding left and right singular vectors $\mathbf{u}_{\min}$ and $\mathbf{v}_{\min}$. $\mathbf{x}^{(0)}$ is determined from* (3.11), *$\mathbf{c} = \mathbf{x}^{(0)}$ and the stopping condition is the same as in the previous examples.*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)},\beta^{(k)},\varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	0.0000e+00	0	3.4855e-03	3.4825e-02	1.0e-01	1.1926e-01
1	1.0137e-01	0	3.5747e-03	5.6058e-03	2.4e-02	-5.9111e-02
2	1.2569e-01	0	6.7098e-04	4.2693e-04	2.2e-03	-3.9088e-02
3	1.2789e-01	0	5.5638e-04	3.2627e-06	1.7e-05	-3.7857e-02
4	1.2790e-01	0	5.5549e-04	1.9673e-10	1.0e-09	-3.7849e-02
5	1.2790e-01	0	5.5549e-04	2.2460e-16	1.2e-15	-3.7849e-02

Table 3.5: Results for Example 3.4.3, $n = 6$. Almost quadratic convergence can be seen in column 6 of the above table.

Table 3.5 shows the results for $n = 6$. In this case, the eigenvalues 7.7080×10^{-2} and 1.8576×10^{-1} closest to zero coalesce at 1.2790×10^{-1} for a value of $\varepsilon = 5.5549 \times 10^{-4}$. This means that $z^* = 1.8576 \times 10^{-1}$ is a double eigenvalue of the nearby defective matrix $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ with, $\varepsilon^* = 5.5549 \times 10^{-4}$ and the computed

value of $\mathbf{x}^* = [\mathbf{u}, \mathbf{v}]^T$, $\mathbf{u}$ *and* $\mathbf{v}$ *are the left and right eigenvectors of* $\mathbf{B}$ *corresponding to the eigenvalue* z^*. *Table 3.6 shows the results for* $n = 12$. *In this case the eigenvalues*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	0.0000e+00	0	1.1186e-08	4.3121e-07	1.8e-02	2.2236e-10
1	1.8010e-02	0	4.3454e-09	1.2695e-07	1.2e-02	-3.9762e-10
2	2.9691e-02	0	1.1251e-09	3.4180e-08	6.4e-03	-5.2812e-11
3	3.6065e-02	0	3.5327e-10	7.2436e-09	2.3e-03	-1.1397e-11
4	3.8343e-02	0	2.0008e-10	7.5968e-10	3.0e-04	-6.3856e-12
5	3.8644e-02	0	1.8521e-10	1.2385e-11	5.1e-06	-5.9627e-12
6	3.8649e-02	0	1.8499e-10	3.6841e-15	1.5e-09	-5.9560e-12
7	3.8649e-02	0	1.8499e-10	6.1460e-17	2.5e-11	-5.9560e-12

Table 3.6: Results of Example 3.4.3, for $n = 12$. Note that we obtained a slower rate of convergence in column 5 in the table above.

3.1028×10^{-2} *and* 4.9509×10^{-2} *closest to zero coalesce at* 3.8649×10^{-2} *for a value of* $\varepsilon = 1.8499 \times 10^{-10}$. *This means that* $z^* = 3.8649 \times 10^{-2}$ *is a double eigenvalue of the nearby defective matrix* $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ *with,* $\varepsilon^* = 1.8499 \times 10^{-10}$ *and the computed value of* $\mathbf{x}^* = [\mathbf{u}, \mathbf{v}]^T$, $\mathbf{u}$ *and* $\mathbf{v}$ *are the left and right eigenvectors of* $\mathbf{B}$ *corresponding to the eigenvalue* z^*.

Again the last columns in tables 3.5 and 3.6 show the values for $F_{\alpha\beta}^{(k)}$. *From Table 3.5 we see that if* $F_{\alpha\beta}^{(k)}$ *is not too small, quadratic, or almost quadratic convergence of Newton's method is obtained in column 6. However, for a small value of* $F_{\alpha\beta}^{(k)}$, *as in Table 3.6, a slower convergence rate is observed.*

Example 3.4.4. *Consider the* 20×20 *bi-diagonal matrix whose diagonal entries are* $20, 19, \ldots, 1$ *and the super-diagonals are* 20. *This matrix was considered by Wilkinson in [62] and has eigenvalues* $1, 2, \ldots, 20$. *Wilkinson has shown that if* ε *is added in position* $(20, 1)$, *then for* $\varepsilon = 10^{-10}$, *the eigenvalues display some sort of symmetry around* 10.5. *As* ε *grows [4] from* 0 *and is approximately equal to* 7.8×10^{-14} *the eigenvalues* 10 *and* 11 *move together and coalesce at* 10.5 *to form a defective eigenvalue.*

We seek ε, u *and* $\mathbf{v}$ *such that* $\mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ *has a defective eigenvalue around* 10.5, *where* $\mathbf{A}$ *is the Wilkinson matrix. As initial guess we take* $\alpha^{(0)} = 10.2$, $\beta^{(0)} = 0$, $\varepsilon^{(0)} = \sigma_{\min}$, $\mathbf{u}^{(0)} = \mathbf{u}_{\min}$ *and* $\mathbf{v}^{(0)} = \mathbf{v}_{\min}$, *where* $\sigma_{\min}$ *is the minimum singular value of* $\mathbf{A}$ *with corresponding left and right singular vectors* $\mathbf{u}_{\min}$ *and* $\mathbf{v}_{\min}$. $\mathbf{x}^{(0)}$

is determined from (3.11) *and* $\mathbf{c} = \mathbf{x}^{(0)}$. *The stopping condition is the same as in Example 3.4.1.*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)}, \beta^{(k)}, \varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	10.2000	0	3.6322e-14	7.7484e-14	4.2e-01	-9.5152e-26
1	10.6194	0	1.0132e-13	4.1197e-14	1.2e-01	-8.5682e-26
2	10.4948	0	6.5737e-14	2.7207e-15	5.2e-03	-8.7808e-26
3	10.5000	0	6.1272e-14	4.0129e-18	2.0e-07	-8.7811e-26
4	10.5000	0	6.1264e-14	9.6368e-27	1.9e-14	-8.7811e-26
5	10.5000	0	6.1264e-14	3.7092e-29	1.1e-16	-8.7811e-26

Table 3.7: Results for Example 3.4.4. Superlinear and quadratic convergence is obtained in columns 5 and 6 respectively.

The numerical results are shown in Table 3.7. We see that β *is zero and* $z = 10.5$ *for a value of* $\varepsilon = 6.1264 \times 10^{-14}$. *Hence ,* $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^H$ *is a defective matrix with defective eigenvalue* $z = 10.5$, *where* $\mathbf{u}$ *and* $\mathbf{v}$ *have been computed within our iteration. We see that the values of* $F_{\alpha\beta}^{(k)}$ *are very small though, given the extremely small value for* $F_{\alpha\beta}^{*}$ *it is surprising that the method even converges, though it was sensitive to the starting guess.*

Note that in theory the values of α *and* β *are real, however, in practice, since both* $\mathbf{v}$ *and* $\mathbf{u}$ *are complex imaginary entries at roundoff level can occur.*

Example 3.4.5. *Let* $\mathbf{A} \in \mathbb{C}^{n\times n}$ *be the Grcar matrix taken from the Matlab gallery* $\mathbf{A} =$ `gallery('grcar',n)`, *where* $n = 6, 20$. *The eigenvalues of* $\mathbf{A}$ *appear in complex conjugate pairs and hence in this case two pairs of complex eigenvalues of* $\mathbf{A}$ *coalesce at two boundary points of the pseudospectrum.*

As initial guesses for $n = 6$ *we take* $\alpha^{(0)} = 0$, $\beta^{(0)} = -1$, $\varepsilon^{(0)} = 0$, $\mathbf{u}^{(0)} = \mathbf{u}_{\min}$ *and* $\mathbf{v}^{(0)} = \mathbf{v}_{\min}$, *where* $\mathbf{u}_{\min}$ *and* $\mathbf{v}_{\min}$ *are left and right singular vectors of* $\mathbf{A} - \beta^{(0)}i\mathbf{I}$, *corresponding to the smallest singular value.* $\mathbf{x}^{(0)}$ *is determined from* (3.11). *The stopping condition is the same as in Example 3.4.1. For* $n = 20$ *we take* $\beta^{(0)} = -2.5$, *the initial guesses for the remaining values are determined similarly. Furthermore* $\mathbf{c} = \mathbf{x}^{(0)}$.

Table 3.8 shows the results for $n = 6$. *The eigenvalue pairs* $1.1391 \pm 1.2303i$ *and* $3.5849 \times 10^{-1} \pm 1.9501i$ *coalesce at* $7.5332 \times 10^{-1} \pm 1.5912i$ *for a value of* $\varepsilon = 2.1519 \times 10^{-1}$.

Table 3.9 shows the results for $n = 20$. *The eigenvalue pairs* $1.0802 \times 10^{-1} \pm$

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)},\beta^{(k)},\varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	0.0000e+00	-1.0000	0.0000e+00	5.0533e-01	2.0e+00	1.4186e-01
1	1.2141e+00	-2.3756	-7.4297e-01	2.2193e+01	1.3e+00	-2.7279e+04
2	1.1159e+00	-1.4291	9.5425e-02	5.2914e-01	6.6e-01	-5.0768e+00
3	1.0512e+00	-1.9848	4.3767e-01	4.1255e-01	5.4e-01	-1.1717e+00
4	8.0543e-01	-1.5940	1.4858e-01	8.6847e-02	8.0e-02	-1.1323e+00
5	7.5742e-01	-1.5944	2.1279e-01	5.5621e-03	5.7e-03	-9.7810e-01
6	7.5335e-01	-1.5912	2.1516e-01	4.2790e-05	4.4e-05	-9.6333e-01
7	7.5332e-01	-1.5912	2.1519e-01	2.4851e-09	2.5e-09	-9.6323e-01
8	7.5332e-01	-1.5912	2.1519e-01	1.6564e-16	2.1e-16	-9.6323e-01

Table 3.8: Results of Example 3.4.5, for $n = 6$. Almost quadratic convergence is shown in columns 5 and 6.

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\varepsilon^{(k)}$	$\|\mathbf{g}(\alpha^{(k)},\beta^{(k)},\varepsilon^{(k)})\|$	$\|\Delta\mathbf{y}^{(k)}\|$	$F_{\alpha\beta}^{(k)}$
0	0.0000e+00	-2.5000	0.0000e+00	1.3806e-01	2.0e-01	9.9103e-01
1	9.5854e-02	-2.3299	1.7989e-02	3.2308e-02	9.5e-02	-2.3623e-01
2	1.3904e-01	-2.2465	1.3564e-03	1.1930e-02	4.8e-02	-1.5963e-01
3	1.6141e-01	-2.2042	7.2914e-04	3.4851e-03	2.3e-02	-2.7982e-02
4	1.5554e-01	-2.1818	4.5435e-04	3.4265e-04	2.2e-03	-2.4693e-02
5	1.5338e-01	-2.1815	4.9060e-04	2.3240e-05	1.5e-04	-2.3956e-02
6	1.5331e-01	-2.1817	4.9141e-04	1.6942e-08	1.1e-07	-2.4012e-02
7	1.5331e-01	-2.1817	4.9141e-04	4.6669e-14	3.1e-13	-2.4012e-02
8	1.5331e-01	-2.1817	4.9141e-04	1.6381e-17	1.0e-16	-2.4012e-02

Table 3.9: Results of Example 3.4.5, for $n = 20$. Columns 5 and 6 shows almost quadratic convergence for $k = 6$ and 7.

$2.2253i$ *and* $2.1882 \times 10^{-1} \pm 2.1132i$ *coalesce at* $1.5331 \times 10^{-1} \pm 2.1817i$ *for a value of* $\varepsilon = 4.9141 \times 10^{-4}$.

The last columns in Tables 3.8-3.9 show the values of $F_{\alpha\beta}^{(k)}$ *which converge to values away from zero. The latter iterates illustrate almost quadratic convergence. Note that in this example* $\beta \neq 0$*, so z is complex, though this makes no difference to the numerical method.*

3.5 Nonlinear System for Finding $\mathrm{d}(\mathbf{A})$ and a Nearby Defective Matrix

In this section, we present an alternative approach to solving the problem considered in this chapter, namely, given a simple matrix $\mathbf{A}$, find a nearby defective matrix $\mathbf{B}$ from $\mathbf{A}$ and the distance between them. This approach involves solving an over-determined system of $2n+3$ nonlinear equations in $2n+2$ unknowns. A method based on the Gauss-Newton theory for computing a nearby defective matrix to $\mathbf{A}$ and the distance between them, which is more efficient than the approach proposed by Alam of Bora [4], even though it does not guarantee that the computed defective matrix is the nearest. In this section, we present the theory only for the case when the nearest defective matrix is real, as in the case for the Examples 3.4.1-3.4.4 in Section 3.4.

We begin by presenting the system of nonlinear equations (3.35), find analytic expressions for the Jacobian and present the key result; Lemma 3.5.2 which shows that the Jacobian is of full rank and so the method should have quadratic convergence with a close enough guess.

As mentioned earlier in the introduction to this chapter, the second approach consists of simply writing down the equations (3.1), (3.2), (3.3) and adding normalisations of the singular vectors. This is the spirit of Newton's method for the standard eigenvalue problem as discussed in Chapter 4 (for the complex case). Thus, we try to solve the following real over-determined system of $(2n+3)$ nonlinear equations

$$\begin{aligned}(\mathbf{A}-z\mathbf{I})\mathbf{v} &= \varepsilon\mathbf{u}\\ \frac{1}{2}\mathbf{v}^T\mathbf{v} &= \frac{1}{2}\\ (\mathbf{A}^T-z\mathbf{I})\mathbf{u} &= \varepsilon\mathbf{v}\\ \frac{1}{2}\mathbf{u}^T\mathbf{u} &= \frac{1}{2}\\ \mathbf{u}^T\mathbf{v} &= 0,\end{aligned} \tag{3.35}$$

in $(2n+2)$ real unknowns: $\mathbf{w} = [\mathbf{v}, \varepsilon, \mathbf{u}, z]^T$. Recall, we have assumed that $\mathbf{u}$ and $\mathbf{v}$ are real, so that all variables in (3.35) are real. After solving for $\mathbf{w}$

in (3.35), we will then compute a nearby defective matrix $\mathbf{B}$ by the formula $\mathbf{B} = \mathbf{A} - \varepsilon \mathbf{u}\mathbf{v}^T$ in line with Alam and Bora [4]. The computed value of ε is then the distance between the simple matrix $\mathbf{A}$ and the defective $\mathbf{B}$.

In compact form $\mathbf{F}(\mathbf{w}) = \mathbf{0}$, (3.35) can be expressed as

$$\mathbf{F}(\mathbf{w}) = \begin{bmatrix} (\mathbf{A} - z\mathbf{I})\mathbf{v} - \varepsilon\mathbf{u} \\ -\frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2} \\ (\mathbf{A}^T - z\mathbf{I})\mathbf{u} - \varepsilon\mathbf{v} \\ -\frac{1}{2}\mathbf{u}^T\mathbf{u} + \frac{1}{2} \\ \mathbf{u}^T\mathbf{v} \end{bmatrix} = \mathbf{0}, \tag{3.36}$$

and the Jacobian of the system of equations (3.36) is given by

$$\mathbf{F}_{\mathbf{w}}(\mathbf{w}) = \begin{bmatrix} (\mathbf{A} - z\mathbf{I}) & -\mathbf{u} & -\varepsilon\mathbf{I} & -\mathbf{v} \\ -\mathbf{v}^T & 0 & \mathbf{0}^T & 0 \\ -\varepsilon\mathbf{I} & -\mathbf{v} & (\mathbf{A}^T - z\mathbf{I}) & -\mathbf{u} \\ \mathbf{0}^T & 0 & -\mathbf{u}^T & 0 \\ \mathbf{u}^T & 0 & \mathbf{v}^T & 0 \end{bmatrix}. \tag{3.37}$$

Let $\mathbf{A} - z\mathbf{I} = \mathbf{U}\Sigma\mathbf{V}^T$ be the singular value decomposition of $\mathbf{A} - z\mathbf{I}$ where

$$\Sigma = \begin{bmatrix} \Sigma_1 & \\ & \varepsilon \end{bmatrix}, \quad \text{with} \quad \Sigma_1 = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_{n-1} \end{bmatrix}, \ \sigma_{n-1} > \varepsilon, \tag{3.38}$$

and $\varepsilon \neq 0$ is a simple smallest singular value of $\mathbf{A} - z\mathbf{I}$. Further, let the Jacobian

$\mathbf{F_w}(\mathbf{w})$ in (3.37) be decomposed as $\mathbf{F_w}(\mathbf{w}) = \mathbf{U_F}\mathbf{G_F}\mathbf{V_F}^T$,

$$\mathbf{F_w}(\mathbf{w}) = \begin{bmatrix} \mathbf{U}\Sigma\mathbf{V}^T & -\mathbf{U}\mathbf{e}_n & -\varepsilon\mathbf{I} & -\mathbf{U}\boldsymbol{\nu} \\ -(\mathbf{V}\mathbf{e}_n)^T & 0 & \mathbf{0}^T & 0 \\ -\varepsilon\mathbf{I} & -\mathbf{V}\mathbf{e}_n & (\mathbf{U}\Sigma\mathbf{V}^T)^T & -\mathbf{V}\boldsymbol{\mu} \\ \mathbf{0}^T & 0 & -(\mathbf{U}\mathbf{e}_n)^T & 0 \\ (\mathbf{V}\boldsymbol{\mu})^T & 0 & (\mathbf{U}\boldsymbol{\nu})^T & 0 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{U} & & & & \\ & 1 & & & \\ & & \mathbf{V} & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix} \begin{bmatrix} \Sigma & -\mathbf{e}_n & -\varepsilon\mathbf{I} & -\boldsymbol{\nu} \\ -\mathbf{e}_n^T & 0 & \mathbf{0}^T & 0 \\ -\varepsilon\mathbf{I} & -\mathbf{e}_n & \Sigma & -\boldsymbol{\mu} \\ \mathbf{0}^T & 0 & -\mathbf{e}_n^T & 0 \\ \boldsymbol{\mu}^T & 0 & \boldsymbol{\nu}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}^T & & & & \\ & 1 & & & \\ & & \mathbf{U}^T & & \\ & & & 1 & \end{bmatrix}, \tag{3.39}$$

where $\mathbf{e}_n$ is the nth column of the n by n identity matrix, $\mathbf{U_F} \in \mathbb{R}^{(2n+3)\times(2n+3)}$, $\mathbf{G_F} \in \mathbb{R}^{(2n+3)\times(2n+2)}$, and $\mathbf{V_F} \in \mathbb{R}^{(2n+2)\times(2n+2)}$. Note that in the above factorization of the Jacobian (3.39), the vectors $\mathbf{u}$ and $\mathbf{v}$ of (3.37) each have two different expressions, *i.e.*,

$$\mathbf{v} = \mathbf{V}\mathbf{e}_n, \quad \text{or} \quad \mathbf{v} = \mathbf{U}\boldsymbol{\nu},$$

and

$$\mathbf{u} = \mathbf{U}\mathbf{e}_n, \quad \text{or} \quad \mathbf{u} = \mathbf{V}\boldsymbol{\mu}.$$

By using the fact that $\mathbf{u}$ and $\mathbf{v}$ are orthogonal *i.e.*, $\mathbf{u}^T\mathbf{v} = 0$, this gives

$$\mathbf{u}^T\mathbf{v} = (\mathbf{U}\mathbf{e}_n)^T\mathbf{U}\boldsymbol{\nu} = \mathbf{e}_n^T(\mathbf{U}^T\mathbf{U})\boldsymbol{\nu} = \mathbf{e}_n^T\boldsymbol{\nu} = \nu_n = 0. \tag{3.40}$$

In the same vein, it can be shown that

$$\mu_n = 0. \tag{3.41}$$

So that at the root, $\boldsymbol{\nu}$ and $\boldsymbol{\mu}$ become respectively

$$\boldsymbol{\nu} = [\boldsymbol{\nu}_{n-1}, 0]^T, \quad \text{and} \quad \boldsymbol{\mu} = [\boldsymbol{\mu}_{n-1}, 0]^T. \tag{3.42}$$

In expanded form, we rewrite the matrix $\mathbf{G_F}$ as

$$\mathbf{G_F} = \begin{bmatrix} \Sigma_1 & \mathbf{0}_{n-1} & \mathbf{0}_{n-1} & -\varepsilon\mathbf{I}_{n-1} & \mathbf{0}_{n-1} & -\boldsymbol{\nu}_{n-1} \\ \mathbf{0}_{n-1}^T & \varepsilon & -1 & \mathbf{0}_{n-1}^T & -\varepsilon & 0 \\ \mathbf{0}_{n-1}^T & -1 & 0 & \mathbf{0}_{n-1}^T & 0 & 0 \\ -\varepsilon\mathbf{I}_{n-1} & \mathbf{0}_{n-1} & \mathbf{0}_{n-1} & \Sigma_1 & \mathbf{0}_{n-1} & -\boldsymbol{\mu}_{n-1} \\ \mathbf{0}_{n-1}^T & -\varepsilon & -1 & \mathbf{0}_{n-1}^T & \varepsilon & 0 \\ \mathbf{0}_{n-1}^T & 0 & 0 & \mathbf{0}_{n-1}^T & -1 & 0 \\ \boldsymbol{\mu}_{n-1}^T & 0 & 0 & \boldsymbol{\nu}_{n-1}^T & 0 & 0 \end{bmatrix}. \tag{3.43}$$

The following preliminary analysis contains some important relationships that will help in proving Lemma 3.5.2 and Lemma 3.5.3 shortly. We build on the assumption that z is a multiple eigenvalue of $\mathbf{B}$ by defining $\hat{\mathbf{v}} = \mathbf{V}\boldsymbol{\beta}$ as the right generalised eigenvector corresponding to the eigenvalue z such that

$$(\mathbf{B} - z\mathbf{I})\hat{\mathbf{v}} = \mathbf{v} \quad \text{with} \quad \mathbf{v}^T\hat{\mathbf{v}} = 0, \quad \text{and} \quad \mathbf{u}^T\hat{\mathbf{v}} \neq 0.$$

The condition $\mathbf{u}^T\hat{\mathbf{v}} \neq 0$ ensures that $\mathbf{B}$ has a 2-dimensional Jordan block only. In the same vein, we define $\hat{\mathbf{u}} = \mathbf{U}\boldsymbol{\alpha}$ as the left generalised eigenvector corresponding to z such that

$$(\mathbf{B}^T - z\mathbf{I})\hat{\mathbf{u}} = \mathbf{u}, \quad \text{with} \quad \mathbf{u}^T\hat{\mathbf{u}} = 0, \quad \text{and} \quad \mathbf{v}^T\hat{\mathbf{u}} \neq 0.$$

By taking transpose of both sides of $(\mathbf{B} - z\mathbf{I})\hat{\mathbf{v}} = \mathbf{v}$, we have $\mathbf{v}^T = \hat{\mathbf{v}}^T(\mathbf{B}^T - z\mathbf{I})$, and post-multiplying by $\hat{\mathbf{u}}$ yields

$$\begin{aligned} \mathbf{v}^T\hat{\mathbf{u}} &= \hat{\mathbf{v}}^T(\mathbf{B}^T - z\mathbf{I})\hat{\mathbf{u}} \\ &= \hat{\mathbf{v}}^T\mathbf{u} \\ &= \mathbf{u}^T\hat{\mathbf{v}}. \end{aligned}$$

This shows that $\mathbf{v}^T\hat{\mathbf{u}} = \mathbf{u}^T\hat{\mathbf{v}}$. The following result now follows.

Lemma 3.5.1. *If $\mathbf{B} = \mathbf{A} - \varepsilon\mathbf{u}\mathbf{v}^T$ has a 2-dimensional Jordan block corresponding to the eigenvalue z, such that $\mathbf{u}^T\hat{\mathbf{v}} \neq 0$, then*

$$\boldsymbol{\mu}_{n-1}^T\Sigma_1^{-1}\boldsymbol{\nu}_{n-1} \neq 0.$$

Proof: Since $\mathbf{B} = \mathbf{A} - \varepsilon \mathbf{u}\mathbf{v}^T$, by post-multiplying both sides of

$$\mathbf{B} - z\mathbf{I} = \mathbf{A} - z\mathbf{I} - \varepsilon \mathbf{u}\mathbf{v}^T,$$

by $\hat{\mathbf{v}}$, and after simplifying we get

$$(\mathbf{B} - z\mathbf{I})\hat{\mathbf{v}} = (\mathbf{A} - z\mathbf{I})\hat{\mathbf{v}}.$$

Similarly, it can be shown that $(\mathbf{B}^T - z\mathbf{I})\hat{\mathbf{u}} = (\mathbf{A}^T - z\mathbf{I})\hat{\mathbf{u}}$. Since $(\mathbf{B} - z\mathbf{I})\hat{\mathbf{v}} = \mathbf{v}$ is the same as $(\mathbf{A} - z\mathbf{I})\hat{\mathbf{v}} = \mathbf{v}$, we have $\mathbf{U}\Sigma(\mathbf{V}^T\mathbf{V})\boldsymbol{\beta} = \mathbf{U}\boldsymbol{\nu}$. Because $\mathbf{U}$, $\mathbf{V}$ are orthogonal, $\Sigma\boldsymbol{\beta} = \boldsymbol{\nu}$ and so

$$\begin{bmatrix} \Sigma_1 & \\ & \varepsilon \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta}_{n-1} \\ \beta_n \end{bmatrix} = \begin{bmatrix} \boldsymbol{\nu}_{n-1} \\ 0 \end{bmatrix}.$$

This means that

$$\boldsymbol{\beta}_{\boldsymbol{n-1}} = \Sigma_1^{-1}\boldsymbol{\nu}_{n-1}.$$

Observe that after premultiplying $\hat{\mathbf{v}} = \mathbf{V}\boldsymbol{\beta}$ by $\mathbf{V}^T$, we have $\boldsymbol{\beta} = \mathbf{V}^T\hat{\mathbf{v}}$. By following the steps that led to (3.40) using $\mathbf{v}^T\hat{\mathbf{v}} = 0$, with $\mathbf{v} = \mathbf{V}\mathbf{e}_n$ and $\hat{\mathbf{v}} = \mathbf{V}\boldsymbol{\beta}$, it can be shown that $\beta_n = 0$. So that we can also write

$$\boldsymbol{\beta}_{\boldsymbol{n-1}} = \mathbf{V}_{n-1}^T\hat{\mathbf{v}}_{n-1} = \Sigma_1^{-1}\boldsymbol{\nu}_{n-1}.$$

Similarly, taking $\hat{\mathbf{u}} = \mathbf{U}\boldsymbol{\alpha}$, and $(\mathbf{A}^T - z\mathbf{I})\hat{\mathbf{u}} = \mathbf{u}$, it can be shown as above that

$$\boldsymbol{\alpha}_{\boldsymbol{n-1}} = \mathbf{U}_{n-1}^T\hat{\mathbf{u}}_{n-1} = \Sigma_1^{-1}\boldsymbol{\mu}_{n-1},$$

and by using the fact that $\mathbf{u}^T\hat{\mathbf{v}} \neq 0$,

$$\mathbf{u}^T\hat{\mathbf{v}} = \boldsymbol{\mu}^T\mathbf{V}^T\mathbf{V}\boldsymbol{\beta} = \boldsymbol{\mu}^T\boldsymbol{\beta} = \boldsymbol{\mu}_{n-1}^T\boldsymbol{\beta}_{n-1} = \boldsymbol{\mu}_{n-1}^T\Sigma_1^{-1}\boldsymbol{\nu}_{n-1}. \tag{3.44}$$

Therefore,

$$\boldsymbol{\mu}_{n-1}^T\Sigma_1^{-1}\boldsymbol{\nu}_{n-1} \neq 0.$$

■

In order to apply the Gauss-Newton method to find the solution to the non-

linear least squares problem (1.41), we need to show that the Jacobian $\mathbf{F}_{\mathbf{w}}(\mathbf{w})$ in (3.37) is of full rank. Before we prove that the Jacobian is of full rank, we define the matrix

$$\mathbf{M} = \begin{bmatrix} \Sigma_1 & -\varepsilon\mathbf{I} & -\boldsymbol{\nu}_{n-1} \\ -\varepsilon\mathbf{I} & \Sigma_1 & -\boldsymbol{\mu}_{n-1} \\ \boldsymbol{\mu}_{n-1}^T & \boldsymbol{\nu}_{n-1}^T & 0 \end{bmatrix}. \tag{3.45}$$

$\mathbf{M}$ is obtained from the expanded form of $\mathbf{G}_{\mathbf{F}}$ in (3.43) by deleting appropriate rows and columns-reason of which will be made clear in the proof of Lemma 3.5.2.

Lemma 3.5.2. *The rank of the Jacobian* $\mathbf{F}_{\mathbf{w}}(\mathbf{w})$ *in* (3.37) *is* $2n+2$ *at the root, (that is,* $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^*)$ *is of full rank) if* $\mathbf{M}$ *is nonsingular.*

Proof: Let $\mathbf{M}$ be nonsingular and $\mathbf{p} = [\mathbf{p}_{n-1}, p_n]^T$, $\mathbf{r} = [\mathbf{r}_{n-1}, r_n]^T$ be nonzero vectors, then we want to show that the rank of $\mathbf{F}_{\mathbf{w}}(\mathbf{w})$ is $2n+2$. The rank of $\mathbf{F}_{\mathbf{w}}(\mathbf{w})$ equals the rank of $\mathbf{G}_{\mathbf{F}}$, because in the decomposition $\mathbf{F}_{\mathbf{w}}(\mathbf{w}) = \mathbf{U}_{\mathbf{F}}\mathbf{G}_{\mathbf{F}}\mathbf{V}_{\mathbf{F}}^T$, the matrices $\mathbf{U}_{\mathbf{F}} \in \mathbb{R}^{(2n+3)\times(2n+2)}$ and $\mathbf{V}_{\mathbf{F}} \in \mathbb{R}^{(2n+2)\times(2n+2)}$ are orthogonal, while $\mathbf{G}_{\mathbf{F}} \in \mathbb{R}^{(2n+3)\times(2n+2)}$ is of the same size as $\mathbf{F}_{\mathbf{w}}(\mathbf{w})$. So it is enough to show that the rank of $\mathbf{G}_{\mathbf{F}}$ equals $2n+2$. This is the same as showing that the $2n+2$ vectors $[\mathbf{p}, q, \mathbf{r}, s]^T$ are zero in

$$\begin{bmatrix} \Sigma & -\mathbf{e}_n & -\varepsilon\mathbf{I} & -\boldsymbol{\nu} \\ -\mathbf{e}_n^T & 0 & \mathbf{0}^T & 0 \\ -\varepsilon\mathbf{I} & -\mathbf{e}_n & \Sigma & -\boldsymbol{\mu} \\ \mathbf{0}^T & 0 & -\mathbf{e}_n^T & 0 \\ \boldsymbol{\mu}^T & 0 & \boldsymbol{\nu}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ q \\ \mathbf{r} \\ s \end{bmatrix} = \mathbf{0}.$$

To make things easier, we will use the expanded form of $\mathbf{G}_{\mathbf{F}}$ in (3.43) in our analysis. Multiply $\mathbf{G}_{\mathbf{F}}$ from the right by the vector $[\mathbf{p}_{n-1}, p_n, q, \mathbf{r}_{n-1}, r_n, s]^T$, upon expanding the matrix vector multiplication and equating to the zero vector on the right hand side, we obtain $p_n = q = r_n = 0$. This is equivalent to deleting the nth, $(n+1)$th, $(2n+1)$th columns and the nth, $(n+1)$th, $(2n+1)$th, $(2n+2)$th rows of $\mathbf{G}_{\mathbf{F}}$, the remaining nonzero entries constitute the

matrix $\mathbf{M}$. Thus, we are left to show that $\mathbf{p}_{n-1} = \mathbf{r}_{n-1} = \mathbf{0}$ and $s = 0$ in

$$\begin{aligned} \Sigma_1 \mathbf{p}_{n-1} - \varepsilon \mathbf{r}_{n-1} - \boldsymbol{\nu}_{n-1} s &= \mathbf{0} \\ -\varepsilon \mathbf{p}_{n-1} + \Sigma_1 \mathbf{r}_{n-1} - \boldsymbol{\mu}_{n-1} s &= \mathbf{0} \\ \boldsymbol{\mu}_{n-1}^T \mathbf{p}_{n-1} + \boldsymbol{\nu}_{n-1}^T \mathbf{r}_{n-1} &= 0, \end{aligned} \tag{3.46}$$

which amounts to showing that $\mathbf{M}$ is nonsingular. But by assumption, $\mathbf{M}$ is nonsingular, hence, $\mathbf{p}_{n-1} = \mathbf{r}_{n-1} = \mathbf{0}$ and $s = 0$. Therefore, $[\mathbf{p}, q, \mathbf{r}, s] = [\mathbf{p}_{n-1}, p_n, q, \mathbf{r}_{n-1}, r_n, s] = \mathbf{0}$ and $\operatorname{rank}(\mathbf{G_F}) = 2n + 2$. Since $\operatorname{rank}(\mathbf{F_w}(\mathbf{w})) = \operatorname{rank}(\mathbf{G_F})$, thus $\operatorname{rank}(\mathbf{F_w}(\mathbf{w})) = 2n + 2$ or the Jacobian is of full rank. ∎

Next, we will make use of Keller's [33] ABCD Lemma 1.3.1 to prove that $\mathbf{M}$ is nonsingular. To do this, we partition $\mathbf{M}$ as follows

$$\mathbf{M} = \begin{bmatrix} \mathsf{A} & \mathbf{b} \\ \mathbf{c}^T & d \end{bmatrix},$$

where in this case

$$\mathsf{A} = \begin{bmatrix} \Sigma_1 & -\varepsilon \mathbf{I} \\ -\varepsilon \mathbf{I} & \Sigma_1 \end{bmatrix}; \quad \mathbf{b} = \begin{bmatrix} -\boldsymbol{\nu}_{n-1} \\ -\boldsymbol{\mu}_{n-1} \end{bmatrix}, \quad \mathbf{c}^T = [\boldsymbol{\mu}_{n-1}^T, \boldsymbol{\nu}_{n-1}^T], \quad \text{and} \quad d = 0. \tag{3.47}$$

One of the cases of the ABCD Lemma [33] (*cf.*, Lemma 1.3.1) for showing that the partitioned matrix $\mathbf{M}$ is nonsingular, is when A is nonsingular. So, we need to show that A is nonsingular. Which is equivalent to showing that $\mathbf{a}$ and $\mathbf{h}$ are both zero vectors in

$$\begin{bmatrix} \Sigma_1 & -\varepsilon \mathbf{I} \\ -\varepsilon \mathbf{I} & \Sigma_1 \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{h} \end{bmatrix} = \mathbf{0}. \tag{3.48}$$

Multiply the first n rows by ε and the second by Σ_1 and after adding, we have

$$\begin{bmatrix} \Sigma_1 & -\varepsilon \mathbf{I} \\ 0\mathbf{I} & \Sigma_1^2 - \varepsilon^2 \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{h} \end{bmatrix} = \mathbf{0}.$$

Using (3.49), the fact that $\Sigma_1^2 - \varepsilon^2 \mathbf{I}$ is nonsingular if $\varepsilon < \sigma_{n-1}$ (*cf.*, (3.38)), accordingly, $\mathbf{h} = \mathbf{0}$. In the same fashion, because Σ_1 is nonsingular we obtain $\mathbf{a} = \mathbf{0}$. Therefore, A is nonsingular.

The following lemma shows that under certain conditions on ε, the matrix $\mathbf{M}$ defined by (3.45) is nonsingular.

Lemma 3.5.3. *Assume that*

$$\varepsilon < \sigma_{n-1}. \tag{3.49}$$

Also, assume ε is so small that its second and higher powers can be neglected, and that

$$\varepsilon < \frac{2\mathbf{u}^T\hat{\mathbf{v}}}{\|\hat{\mathbf{u}}_{n-1}\|^2 + \|\hat{\mathbf{v}}_{n-1}\|^2}, \tag{3.50}$$

then the matrix $\mathbf{M}$ is nonsingular.

Proof: Since A has been shown to be nonsingular, in establishing the nonsingularity of $\mathbf{M}$ using the ABCD Lemma, all we need is to show that the Schur complement, $d - \mathbf{c}^T\mathsf{A}^{-1}\mathbf{b}$, is not equal to zero, where $\mathbf{b}$, $\mathbf{c}^T$ and d are as defined in (3.47). To begin, we note that $\mathsf{A}^{-1}\mathbf{b}$ is the same as solving for $[\mathbf{f}, \mathbf{g}]^T$ in

$$\begin{bmatrix} \Sigma_1 & -\varepsilon\mathbf{I} \\ -\varepsilon\mathbf{I} & \Sigma_1 \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{\nu}_{n-1} \\ -\boldsymbol{\mu}_{n-1} \end{bmatrix}. \tag{3.51}$$

Multiply the first n-equations by ε and the second one by Σ_1, add them together, solve for $\mathbf{g}$ to obtain

$$\mathbf{g} = -[\Sigma_1^2 - \varepsilon^2\mathbf{I}]^{-1}(\Sigma_1\boldsymbol{\mu}_{n-1} + \varepsilon\boldsymbol{\nu}_{n-1}). \tag{3.52}$$

Hence, by substituting $\mathbf{g}$ into $\boldsymbol{\Sigma}_1\mathbf{f} - \varepsilon\mathbf{g} = -\boldsymbol{\nu}_{n-1}$; we obtain

$$\mathbf{f} = \varepsilon\boldsymbol{\Sigma}_1^{-1}\mathbf{g} - \boldsymbol{\Sigma}_1^{-1}\boldsymbol{\nu}_{n-1}.$$

Which is equivalent to

$$\mathbf{f} = -\left\{\varepsilon\Sigma_1^{-1}[\Sigma_1^2 - \varepsilon^2\mathbf{I}]^{-1}(\Sigma_1\boldsymbol{\mu}_{n-1} + \varepsilon\boldsymbol{\nu}_{n-1}) + \Sigma_1^{-1}\boldsymbol{\nu}_{n-1}\right\}. \tag{3.53}$$

So that

$$\begin{aligned} d - \mathbf{c}^T \mathsf{A}^{-1}\mathbf{b} &= -[\boldsymbol{\mu}_{n-1}^T \quad \boldsymbol{\nu}_{n-1}^T] \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix} \\ &= \varepsilon \boldsymbol{\mu}_{n-1}^T \Sigma_1^{-1} [\Sigma_1^2 - \varepsilon^2 \mathbf{I}]^{-1} (\Sigma_1 \boldsymbol{\mu}_{n-1} + \varepsilon \boldsymbol{\nu}_{n-1}) + \boldsymbol{\mu}_{n-1}^T \Sigma_1^{-1} \boldsymbol{\nu}_{n-1} \\ &+ \boldsymbol{\nu}_{n-1}^T [\Sigma_1^2 - \varepsilon^2 \mathbf{I}]^{-1} (\Sigma_1 \boldsymbol{\mu}_{n-1} + \varepsilon \boldsymbol{\nu}_{n-1}). \end{aligned} \tag{3.54}$$

From the statement of the Lemma, if ε is so small that its second and higher powers can be neglected, with some simplifications, we have

$$\begin{aligned} d - \mathbf{c}^T \mathsf{A}^{-1}\mathbf{b} &= 2\boldsymbol{\mu}_{n-1}^T \Sigma_1^{-1} \boldsymbol{\nu}_{n-1} + \varepsilon (\boldsymbol{\mu}_{n-1}^T \Sigma_1^{-2} \boldsymbol{\mu}_{n-1} + \boldsymbol{\nu}_{n-1}^T \Sigma_1^{-2} \boldsymbol{\nu}_{n-1}) \\ &= 2\boldsymbol{\mu}_{n-1}^T \Sigma_1^{-1} \boldsymbol{\nu}_{n-1} + \varepsilon [(\Sigma_1^{-1} \boldsymbol{\mu}_{n-1})^T (\Sigma_1^{-1} \boldsymbol{\mu}_{n-1}) \\ &+ (\Sigma_1^{-1} \boldsymbol{\nu}_{n-1})^T (\Sigma_1^{-1} \boldsymbol{\nu}_{n-1})] \\ &= 2\boldsymbol{\mu}_{n-1}^T \Sigma_1^{-1} \boldsymbol{\nu}_{n-1} + \varepsilon (\|\Sigma_1^{-1} \boldsymbol{\mu}_{n-1}\|^2 + \|\Sigma_1^{-1} \boldsymbol{\nu}_{n-1}\|^2). \end{aligned} \tag{3.55}$$

It remains to be shown that the expression on the right side of (3.55) is nonzero provided (3.49) and (3.50) holds. With the simplified expression,

$$\mathbf{u}^T \hat{\mathbf{v}} = \boldsymbol{\mu}_{n-1}^T \Sigma_1^{-1} \boldsymbol{\nu}_{n-1} \neq 0,$$

in (3.44), (3.55) now becomes

$$\begin{aligned} d - \mathbf{c}^T \mathsf{A}^{-1}\mathbf{b} &= 2\mathbf{u}^T \hat{\mathbf{v}} + \varepsilon (\|\mathbf{U}_{n-1}^T \hat{\mathbf{u}}_{n-1}\|^2 + \|\mathbf{V}_{n-1}^T \hat{\mathbf{v}}_{n-1}\|^2) \\ &= 2\mathbf{u}^T \hat{\mathbf{v}} + \varepsilon (\|\hat{\mathbf{u}}_{n-1}\|^2 + \|\hat{\mathbf{v}}_{n-1}\|^2). \end{aligned} \tag{3.56}$$

In arriving at the expression on the right hand side above, we made use of the fact that the matrices $\mathbf{V}_{n-1}$ and $\mathbf{U}_{n-1}$ are orthogonal. Therefore, $d - \mathbf{c}^T \mathsf{A}^{-1}\mathbf{b}$ is not equal to zero, if for small enough ε,

$$\varepsilon < \frac{2\mathbf{u}^T \hat{\mathbf{v}}}{\|\hat{\mathbf{u}}_{n-1}\|^2 + \|\hat{\mathbf{v}}_{n-1}\|^2}.$$

This shows that the matrix $\mathbf{M}$ is nonsingular if (3.49) and (3.50) holds. ∎

Note that the Jacobian is of full rank under the conditions in which the ma-

trix $\mathbf{M}$ is nonsingular. Since we have established that the Jacobian is of full rank, we can conveniently seek a solution for $\Delta\mathbf{w}^{(k)}$ in (1.46). If we apply the Gauss-Newton method discussed in Subsection 1.5 of Chapter 1 to the over-determined nonlinear system (3.36), then we obtain the following (*cf.* (1.47) and (1.48))

$$\mathbf{R}\Delta\mathbf{w}^{(k)} = -\mathbf{Q}^T\mathbf{F}(\mathbf{w}^{(k)}), \quad \text{and} \quad \mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \Delta\mathbf{w}^{(k)},$$

where in this case $\mathbf{F}(\mathbf{w}^{(k)}) = \mathbf{QR}$, $\mathbf{Q} \in \mathbb{R}^{(2n+3)\times(2n+2)}$ and $\mathbf{R} \in \mathbb{R}^{(2n+2)\times(2n+2)}$.

Now, we present a Gauss-Newton based algorithm for finding the parameters for computing a nearby defective matrix to a simple matrix. This is presented in Algorithm 13.

Algorithm 13 Gauss-Newton Algorithm for Computing a Nearby Defective Matrix

Input: $\mathbf{w}^{(0)} = [\mathbf{v}^{(0)}, \varepsilon^{(0)}, \mathbf{u}^{(0)}, z^{(0)}]^T$, k_{max} and tol.

1: **for** $k = 0, 1, 2, \ldots$ until convergence **do**
2: Find the reduced QR factorization of $\mathbf{F}_{\mathbf{w}}(\mathbf{w}^{(k)})$ in (3.37).
3: Compute the matrix-vector multiplication $\mathbf{y}^{(k)} = -\mathbf{Q}^T\mathbf{F}(\mathbf{w}^{(k)})$.
4: Solve the upper-triangular system $\mathbf{R}\Delta\mathbf{w}^{(k)} = \mathbf{y}^{(k)}$ for $\Delta\mathbf{w}^{(k)}$.
5: Update, $\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \Delta\mathbf{w}^{(k)}$.
6: **end for**

Output: $\mathbf{w}^*$.

Algorithm 13 should be stopped as soon as the norm of $\Delta\mathbf{w}^{(k)}$ is less than or equal to some user defined tolerance.

In the next section, we present the result of numerical experiments which confirms the theory.

3.6 Numerical Experiments

In this section, we present result of numerical experiments which confirms the theory discussed in the last section. We show that Algorithm 13 works for the Trefethen, Kahan, Frank and Wilkinson matrices. The results obtained agrees with those of Section 3.4 using Algorithm 12 with the same starting guesses.

Example 3.6.1. *Consider the matrix* $\mathbf{A} = \begin{bmatrix} -1 & 5 \\ 0 & -2 \end{bmatrix}$, *(see [61]). As initial guesses we choose* $\alpha^{(0)} = \beta^{(0)} = 0$, $\varepsilon^{(0)} = \sigma_{\min}$, $\mathbf{u}^{(0)} = \mathbf{u}_{\min}$ *and* $v^{(0)} = \mathbf{v}_{\min}$, *where* $\sigma_{\min}$ *is the minimum singular value of* $\mathbf{A}$ *with corresponding left and right singular vectors* $\mathbf{u}_{\min}$ *and* $\mathbf{v}_{\min}$. *We stop the iteration once*

$$\|\Delta \mathbf{w}^{(k)}\| < \tau, \quad \textit{where} \quad \tau = 8 \times 10^{-16}.$$

Table 3.10 shows the results for Example 3.4.1. Hence, $z = -1.5$ *with* $\varepsilon = 4.9510 \times$

k	$\alpha^{(k)}$	$\varepsilon^{(k)}$	$\lvert\varepsilon^{(k+1)} - \varepsilon^{(k)}\rvert$	$\lvert\alpha^{(k+1)} - \alpha^{(k)}\rvert$	$\|\mathbf{F}(\mathbf{w}^{(k)})\|$	$\|\Delta \mathbf{w}^{(k)}\|$
0	0.0000	3.6597e-1	1.0e+00	2.0e+00	5.1e-01	2.3e+00
1	-2.0400	6.8361e-1	6.0e-01	5.0e-01	1.2e+00	7.8e-01
2	-1.5386	8.5612e-2	3.6e-02	3.8e-02	7.0e-02	5.3e-02
3	-1.5001	4.9585e-2	7.6e-05	8.1e-05	1.5e-04	1.1e-04
4	-1.5000	4.9510e-2	1.6e-10	1.7e-10	3.3e-10	2.3e-10
5	-1.5000	4.9510e-2	6.9e-18	0.0e+00	2.2e-17	9.0e-18

Table 3.10: Columns five and six shows quadratic convergence for Example 3.4.1. Quadratic convergence is lost in the last row, possibly due to round off errors.

10^{-2}, $\mathbf{u} = [-9.8538 \times 10^{-2}, -9.9513 \times 10^{-1}]^T$ *and* $\mathbf{v} = [9.9513 \times 10^{-1}, -9.8538 \times 10^{-2}]^T$. *Therefore,* $\mathbf{B} = \mathbf{A} - \varepsilon \mathbf{u}\mathbf{v}^H$ *is a defective matrix. The method converges quadratically in* 6 *iterations, as expected from Newton's method. The computed values of* z^* *and* ε^* *agree with those of Table 3.1.*

Example 3.6.2. *Let* $\mathbf{A} \in \mathbb{C}^{n \times n}$ *be the Kahan matrix [61], which is given by* (3.34). *We consider this matrix for* $n = 6, 15, 20$. *The starting values and stopping condition are chosen as in Example 3.4.1. Table 3.6.2 shows the results for* $n = 6$. *In this case, the eigenvalues* 1.5849×10^{-1} *and* 10^{-1} *coalesce at* 1.2763×10^{-1} *for a value of* $\varepsilon = 4.7049 \times 10^{-4}$. *Quadratic convergence rate is observed in rows six and seven of column five, and the computed values of* z^* *and* ε^* *agree with those of Table 3.2.*

Table 3.12 shows the results for $n = 15$. *In this case, the eigenvalues* 1.1788×10^{-1} *and* 10^{-1} *coalesce at* 1.0729×10^{-1} *for a value of* $\varepsilon = 5.1757 \times 10^{-7}$. *The computed values of* z^* *and* ε^* *agree with those of Table 3.3.*

Table 3.13 shows the results for $n = 20$. *In this case, the eigenvalues* 1.1288×10^{-1} *and* 10^{-1} *coalesce at* 1.0501×10^{-1} *for a value of* $\varepsilon = 2.8841 \times 10^{-8}$. *The*

k	$\alpha^{(k)}$	$\varepsilon^{(k)}$	$\lvert\varepsilon^{(k+1)} - \varepsilon^{(k)}\rvert$	$\lvert\alpha^{(k+1)} - \alpha^{(k)}\rvert$	$\lVert\mathbf{F}(\mathbf{w}^{(k)})\rVert$	$\lVert\Delta\mathbf{w}^{(k)}\rVert$
0	0.0000e+0	9.9694e-03	2.2e-02	1.4e-01	1.6e-01	3.6e-01
1	1.3643e-1	1.2145e-2	1.2e-02	2.0e-02	7.1e-02	2.2e-01
2	1.1639e-1	3.7277e-4	6.1e-05	9.5e-03	2.5e-02	5.6e-02
3	1.2590e-1	4.3373e-4	3.7e-05	1.7e-03	1.5e-03	1.3e-02
4	1.2760e-1	4.7082e-4	3.3e-07	3.2e-05	8.9e-05	3.5e-04
5	1.2763e-1	4.7049e-4	2.9e-10	1.3e-08	6.3e-08	1.5e-07
6	1.2763e-1	4.7049e-4	4.4e-17	2.2e-15	1.1e-14	2.2e-14
7	1.2763e-1	4.7049e-4	5.4e-20	0.0e+00	9.8e-17	6.8e-17

Table 3.11: Results for Example 3.4.2, $n = 6$ using Algorithm 13. We observe quadratic convergence in the last column.

k	$\alpha^{(k)}$	$\varepsilon^{(k)}$	$\lvert\varepsilon^{(k+1)} - \varepsilon^{(k)}\rvert$	$\lvert\alpha^{(k+1)} - \alpha^{(k)}\rvert$	$\lVert\mathbf{F}(\mathbf{w}^{(k)})\rVert$	$\lVert\Delta\mathbf{w}^{(k)}\rVert$
0	0.0000e+0	4.7454e-4	7.3e-04	5.9e-02	1.2e-02	2.0e-01
1	5.9261e-2	2.5746e-4	2.6e-04	2.5e-02	2.1e-02	2.2e-01
2	8.4526e-2	4.7975e-7	8.7e-07	1.3e-02	2.5e-02	1.7e-01
3	9.7390e-2	1.3523e-6	2.0e-06	7.2e-03	1.5e-02	1.1e-01
4	1.0455e-1	6.6553e-7	1.3e-07	2.4e-03	6.3e-03	4.4e-02
5	1.0699e-1	5.3550e-7	1.8e-08	3.0e-04	9.7e-04	5.7e-03
6	1.0728e-1	5.1778e-7	2.0e-10	4.3e-06	1.6e-05	8.3e-05
7	1.0729e-1	5.1757e-7	4.1e-14	8.9e-10	3.4e-09	1.7e-08
8	1.0729e-1	5.1757e-7	1.6e-21	2.8e-17	1.8e-16	7.0e-16

Table 3.12: Results for Example 3.4.2, $n = 15$ using Algorithm 13. Quadratic convergence is observed in rows eight and nine of the last column.

k	$\alpha^{(k)}$	$\varepsilon^{(k)}$	$\lvert\varepsilon^{(k+1)} - \varepsilon^{(k)}\rvert$	$\lvert\alpha^{(k+1)} - \alpha^{(k)}\rvert$	$\lVert\mathbf{F}(\mathbf{w}^{(k)})\rVert$	$\lVert\Delta\mathbf{w}^{(k)}\rVert$
0	0.0000e+00	1.3141e-04	1.9e-04	4.7e-02	4.0e-03	1.7e-01
1	4.7216e-02	5.7554e-05	5.8e-05	2.7e-02	1.4e-02	1.9e-01
2	7.3921e-02	3.1613e-08	4.3e-07	1.5e-02	1.9e-02	1.8e-01
3	8.9281e-02	4.0206e-07	4.5e-07	9.3e-03	1.6e-02	1.4e-01
4	9.8568e-02	5.2530e-08	1.5e-08	4.7e-03	1.0e-02	9.0e-02
5	1.0329e-01	3.7362e-08	7.9e-09	1.5e-03	4.0e-03	3.3e-02
6	1.0483e-01	2.9487e-08	6.4e-10	1.8e-04	5.4e-04	3.9e-03
7	1.0501e-01	2.8848e-08	7.3e-12	2.2e-06	7.5e-06	4.8e-05
8	1.0501e-01	2.8841e-08	1.1e-15	3.3e-10	1.2e-09	7.3e-09
9	1.0501e-01	2.8841e-08	3.3e-23	0.0e+00	1.9e-16	2.3e-16

Table 3.13: Results for Example 3.4.2, for $n = 20$ using Algorithm 13. We observed almost quadratic convergence in the last two rows of the last column.

computed values of z^* and ε^* agree with those of Table 3.4.

Example 3.6.3. *Let* $\mathbf{A} \in \mathbb{C}^{n\times n}$ *be the Frank matrix taken from the Matlab gallery* $\mathbf{A}$ = `gallery('frank',n)`*, where* $n = 6, 12$*. As initial guesses we choose* $\alpha^{(0)} = \beta^{(0)} = 0$*,* $\varepsilon^{(0)} = \sigma_{\min}$*,* $\mathbf{u}^{(0)} = \mathbf{u}_{\min}$ *and* $\mathbf{v}^{(0)} = \mathbf{v}_{\min}$*, where* $\sigma_{\min}$ *is the minimum singular value of* $\mathbf{A}$ *with corresponding left and right singular vectors* $\mathbf{u}_{\min}$ *and* $\mathbf{v}_{\min}$*.* $\mathbf{x}^{(0)}$ *is determined from* (3.11)*, the stopping condition is the same as in the previous examples. Table 3.14 shows the results for* $n = 6$*. In this case the eigenvalues* 7.7080×10^{-2} *and* 1.8576×10^{-1} *closest to zero coalesce at* 1.2790×10^{-1} *for a value of* $\varepsilon = 5.5549 \times 10^{-4}$*. The computed values of* z^* *and* ε^* *agree with those of Table 3.5.*

k	$\alpha^{(k)}$	$\varepsilon^{(k)}$	$\lvert\varepsilon^{(k+1)} - \varepsilon^{(k)}\rvert$	$\lvert\alpha^{(k+1)} - \alpha^{(k)}\rvert$	$\lVert\mathbf{F}(\mathbf{w}^{(k)})\rVert$	$\lVert\Delta\mathbf{w}^{(k)}\rVert$
0	0.0000e+00	3.4855e-03	7.1e-03	1.0e-01	7.0e-02	1.6e-01
1	1.0137e-01	3.5747e-03	3.0e-03	2.4e-02	1.4e-02	4.2e-02
2	1.2532e-01	6.0255e-04	4.7e-05	2.6e-03	1.1e-03	4.1e-03
3	1.2788e-01	5.5588e-04	3.9e-07	2.4e-05	1.0e-05	3.9e-05
4	1.2790e-01	5.5549e-04	3.6e-11	2.1e-09	8.8e-10	3.4e-09
5	1.2790e-01	5.5549e-04	1.1e-19	1.7e-16	7.0e-16	3.9e-16

Table 3.14: Results for Example 3.4.3, $n = 6$ using Algorithm 13. We observe almost quadratic convergence in the last column.

k	$\alpha^{(k)}$	$\varepsilon^{(k)}$	$\lvert\varepsilon^{(k+1)} - \varepsilon^{(k)}\rvert$	$\lvert\alpha^{(k+1)} - \alpha^{(k)}\rvert$	$\lVert\mathbf{F}(\mathbf{w}^{(k)})\rVert$	$\lVert\Delta\mathbf{w}^{(k)}\rVert$
0	0.0000e+00	1.1186e-08	1.6e-08	1.8e-02	8.6e-07	2.7e-02
1	1.8010e-02	4.3454e-09	3.9e-09	1.3e-02	4.1e-04	1.9e-02
2	3.0849e-02	4.1641e-10	2.0e-10	6.2e-03	2.1e-04	9.4e-03
3	3.7032e-02	2.2016e-10	3.3e-11	1.5e-03	5.0e-05	2.3e-03
4	3.8559e-02	1.8668e-10	1.7e-12	9.0e-05	3.1e-06	1.4e-04
5	3.8649e-02	1.8499e-10	5.4e-15	3.0e-07	1.1e-08	4.6e-07
6	3.8649e-02	1.8499e-10	1.4e-17	2.6e-11	1.2e-13	3.9e-11
7	3.8649e-02	1.8499e-10	9.0e-18	2.7e-11	1.4e-15	4.1e-11
8	3.8649e-02	1.8499e-10	3.8e-17	7.3e-13	3.5e-16	1.1e-12

Table 3.15: Results for Example 3.4.3, $n = 12$ using Algorithm 13. We observe superlinear convergence in the second to the last column.

Table 3.15 shows the results for $n = 12$*. In this case the eigenvalues* 3.1028×10^{-2} *and* 4.9509×10^{-2} *closest to zero coalesce at* 3.8649×10^{-2} *for a value of* $\varepsilon = 1.8499 \times 10^{-10}$*. The computed values of* z^* *and* ε^* *agree with those of Table 3.6.*

Example 3.6.4. *Consider the* 20×20 *bi-diagonal matrix whose diagonal entries are* $20, 19, \ldots, 1$ *and the super-diagonals are* 20*. This matrix was considered by Wilkinson*

[62] and also in Example 3.4.4 and has eigenvalues $1, 2, \ldots, 20$*. We repeated the same example with the same starting guesses as in Example 3.4.4 but in this case with Algorithm 13. Results are as tabulated in Table 3.16. The computed values of* z^* *and* ε^* *agree with those of Table 3.7.*

k	$\alpha^{(k)}$	$\varepsilon^{(k)}$	$\lvert\varepsilon^{(k+1)} - \varepsilon^{(k)}\rvert$	$\lvert\alpha^{(k+1)} - \alpha^{(k)}\rvert$	$\lVert\mathbf{F}(\mathbf{w}^{(k)})\rVert$	$\lVert\Delta\mathbf{w}^{(k)}\rVert$
0	10.200	3.6322e-14	6.5e-14	4.2e-01	1.6e-13	4.2e-01
1	10.619	1.0152e-13	3.9e-14	1.0e-01	1.5e-02	1.0e-01
2	10.519	6.2701e-14	1.4e-15	1.9e-02	8.5e-04	1.9e-02
3	10.500	6.1312e-14	4.7e-17	1.9e-04	3.0e-05	1.9e-04
4	10.500	6.1264e-14	4.6e-21	6.9e-08	3.0e-09	6.9e-08
5	10.500	6.1264e-14	5.7e-28	4.8e-12	2.6e-15	4.8e-12
6	10.500	6.1264e-14	1.3e-29	0.0e+00	1.9e-15	1.0e-16

Table 3.16: Results for Example 3.4.4 using Algorithm 13. We observed quadratic convergence except for the last two rows of the last column.

3.7 Conclusion

We have developed two new algorithms for computing a nearby defective matrix. Numerical examples show that these new techniques perform well and give quadratic convergence in the generic cases. Also, since the first algorithm is based on Newton's method applied to a real 3-dimensional nonlinear system (with only one LU factorisation required at each step) it is simple to apply and is significantly faster than the technique in [4]. The second algorithm is based on the Gauss-Newton method for computing a nearby defective matrix from a simple one and the distance between them.

However, as has already been mentioned, since the two algorithms are based on Newton's method or its variant, convergence to the nearest defective matrix cannot be guaranteed, though in fact, in all the examples considered, convergence to the nearest defective matrix was achieved. Of course, a more sophisticated nonlinear solver, *e.g.*, global Newton's method or a global minimiser, could be applied to (3.15) if required.

Though Algorithm 12 is designed to compute a nearby defective matrix in the generic case (that is, there is a well-conditioned 2-dimensional Jordan block), the first algorithm has two features that enable it to recognise when

the conditions of Assumption 3.1.1 fail. First, if there is another singular value near ϵ then the condition number of $\mathbf{M}(\alpha, \beta, \varepsilon)$ will be large. Second, if the condition number of $\mathbf{M}(\alpha, \beta, \varepsilon)$ is small, but $F_{\alpha\beta}$ is close to zero at the root, then this indicates the presence of a nearby defective matrix with a Jordan block of dimension greater than 2. As such the algorithm in this chapter could be used to provide starting values for an alternative algorithm that could detect a higher order singularity.

For Algorithm 13, near a three-dimensional Jordan block, the right hand side condition on ε in (3.50) tends to zero, so ε is forced to zero. This means that some diagonal elements of R in $R\Delta\mathbf{w}^{(k)} = -\mathbf{Q}^T\mathbf{F}(\mathbf{w}^{(k)})$ could be small if the Jordan block is of dimension three or σ_{n-1} is close to $\varepsilon = \sigma_n$, and it is not possible to distinguish between these two situations.

CHAPTER 4

Inverse Iteration with a Complex Shift

4.1 Introduction

Let $\mathbf{A}$ be a large sparse, real n by n nonsymmetric matrix and $\mathbf{B} \in \mathbb{R}^{n\times n}$ a symmetric positive definite matrix. In this chapter, we consider the problem of computing the eigenpair $(\mathbf{z}, \lambda)$ from the following generalised complex eigenvalue problem

$$\mathbf{A}\mathbf{z} = \lambda\mathbf{B}\mathbf{z}, \quad \mathbf{z} \in \mathbb{C}^n, \ \ \mathbf{z} \neq \mathbf{0}, \tag{4.1}$$

where $\lambda \in \mathbb{C}$ is the eigenvalue of the pencil $(\mathbf{A}, \mathbf{B})$ and $\mathbf{z}$ its corresponding complex eigenvector. We assume that the eigenpair of interest $(\mathbf{z}, \lambda)$ is algebraically simple, so that ψ^H the corresponding left eigenvector is such that [57, p. 136]

$$\psi^H\mathbf{B}\mathbf{z} \neq 0. \tag{4.2}$$

By adding the normalisation

$$\mathbf{z}^H\mathbf{B}\mathbf{z} = 1, \tag{4.3}$$

to (4.1) and with $\mathbf{v} = [\mathbf{z}^T, \lambda]$, the combined system of equations can be expressed in the form $\mathbf{F}(\mathbf{v}) = \mathbf{0}$ as

$$\mathbf{F}(\mathbf{v}) = \begin{bmatrix} (\mathbf{A} - \lambda\mathbf{B})\mathbf{z} \\ -\frac{1}{2}\mathbf{z}^H\mathbf{B}\mathbf{z} + \frac{1}{2} \end{bmatrix} = \mathbf{0}. \tag{4.4}$$

Note that $\mathbf{z}^H\mathbf{B}\mathbf{z}$ is real since $\mathbf{B}$ is symmetric and positive definite. This results in solving a system of n complex and one real nonlinear equation for the $(n+1)$ complex unknowns $\mathbf{v} = [\mathbf{z}, \lambda]^T$. Note that, if $\mathbf{z}$ from $(\mathbf{z}, \lambda)$ solves (4.4), then so does $e^{i\theta}\mathbf{z}$ for any $\theta \in [0, 2\pi)$. Hence, (4.4) does not have a unique solution. Another drawback of the normalisation (4.3) is that $\bar{\mathbf{z}}$ in $\mathbf{z}^H\mathbf{B}\mathbf{z} = \bar{\mathbf{z}}^T\mathbf{B}\mathbf{z}$ is not differentiable[1]. Therefore, we cannot just differentiate (4.4) and apply the standard Newton's method. In this chapter, we shall show how these drawbacks can be overcome, at least for the $\mathbf{B} = \mathbf{I}$ case.

Recall that for a **real** eigenpair $(\mathbf{z}, \lambda)$, (4.4) gives $(n+1)$ real equations for $(n+1)$ real unknowns and Newton's method for solving (4.4) involves the solution of the $(n+1)$ square linear systems

$$\begin{bmatrix} \mathbf{A} - \lambda^{(k)}\mathbf{B} & -\mathbf{B}\mathbf{z}^{(k)} \\ -(\mathbf{B}\mathbf{z}^{(k)})^T & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{z}^{(k)} \\ \Delta\lambda^{(k)} \end{bmatrix} = - \begin{bmatrix} (\mathbf{A} - \lambda^{(k)}\mathbf{B})\mathbf{z}^{(k)} \\ -\frac{1}{2}{\mathbf{z}^{(k)}}^T\mathbf{B}\mathbf{z}^{(k)} + \frac{1}{2} \end{bmatrix}, \tag{4.5}$$

for the $(n+1)$ real unknowns $\Delta\mathbf{v}^{(k)} = [{\Delta\mathbf{z}^{(k)}}^T, \Delta\lambda^{(k)}]$, and updating $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta\mathbf{v}^{(k)}$ for $k = 0, 1, 2, \ldots$. Secondly, for $(\mathbf{z}, \lambda)$ complex, Ruhe [51] added the normalisation $\mathbf{c}^H\mathbf{z} = 1$, where $\mathbf{c}$ is a fixed complex vector instead of (4.3), so that (4.1) and $\mathbf{c}^H\mathbf{z} = 1$ provide $(n+1)$ complex equations for $(n+1)$ complex unknowns, and the Jacobian of this system is

$$\begin{bmatrix} (\mathbf{A} - \lambda\mathbf{B}) & -\mathbf{B}\mathbf{z} \\ \mathbf{c}^H & 0 \end{bmatrix}.$$

The above Jacobian is square and can be easily shown to be nonsingular, using the ABCD Lemma if the eigenvalue of interest is algebraically simple and $\mathbf{c}^H\mathbf{z} \neq 0$ at the root. One major distinction between our normalisation and Ruhe's is that, ours is the natural normalisation for an eigenvector and we do not worry about how to choose $\mathbf{c}$.

Our approach for analysing the solution of (4.4) for $\mathbf{v}$ begins by splitting the eigenpair $(\mathbf{z}, \lambda)$ into their real and imaginary parts: $\mathbf{z} = \mathbf{z}_1 + i\mathbf{z}_2$, $\lambda = \alpha + i\beta$ where $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^n$, and $\alpha, \beta \in \mathbb{R}$. After expanding (4.4), we obtain

[1] For a single variable, if $z = x + iy$, $\bar{z} = x - iy$, then the Cauchy-Riemann equations are not satisfied because, with $u(x, y) = x$, $v(x, y) = -y$, then $u_x(x, y) = 1$ and $v_y(x, y) = -1$, whereas the Cauchy-Riemann equations (see, for example [35]) require that $u_x(x, y) = v_y(x, y)$. This shows that $\bar{z}$ is not differentiable at (x, y).

a real system of $(2n+1)$ under-determined nonlinear equations in $(2n+2)$ real unknowns $\mathbf{v} = [\mathbf{z}_1, \mathbf{z}_2, \alpha, \beta]^T$, and it is natural to use the Gauss-Newton method (see, for example, Deuflhard [17, pp. 222-223]) to obtain a solution. By linearising the system of under-determined nonlinear equations, we obtain a system of under-determined linear equations involving the corresponding Jacobian. The key results in this chapter are Theorems 4.2.1, 4.4.1 and 4.5.1.

This chapter is structured as follows. In Section 4.2, we show that for an algebraically simple eigenvalue, the Jacobian is of full rank at the root with a known nullvector. Section 4.3 provides theoretical expressions for the exact nullvector of the Jacobian. In Section 4.4, we consider the case $\mathbf{B} = \mathbf{I}$, where we show that by adding an extra equation to the system of under-determined linear equations, one obtains a square one and prove that the solution obtained by solving this square system is equivalent to that obtained by solving the under-determined linear system. The extra equation that will be added, stems from the orthogonality of a known approximate nullvector and the minimum norm solution to the under-determined system of linear equations. In Theorem 4.5.1, we show that the $(2n+2)$ square system of equations is equivalent to the corresponding $(n+1)$ square system (4.5) with $\mathbf{z}$ and λ complex.

To summarise, we give a rigorous proof that, if we ignore the non uniqueness of solution of (4.4) and the fact that (4.3) is not differentiable, and proceed by applying Newton's method to (4.4) formally, then we obtain exactly the same results obtained using the Gauss-Newton method. Computationally, this means we may solve square systems like (4.5) using Gaussian elimination rather than solving rectangular systems as is the case if the Gauss-Newton method were used. The case $\mathbf{B} \neq \mathbf{I}$ is not so nice and as far as we can tell, a similar result does not apply. This is explained further in Section 4.6.

The analysis in each section is supported by a numerical example. All approaches described in this chapter, give quadratic convergence, though, as usual, they rely on good initial guesses to the desired eigenpair. In conclusion, we show the mathematical equivalence of three methods-which is our main aim in this chapter. Throughout this chapter, $\|.\| = \|.\|_2$.

4.2 Computation of Complex Eigenpairs by solving an Under-determined System of Nonlinear Equations

In this section, we will expand the system of n complex and one real nonlinear equations in $(n+1)$ complex unknowns (4.4) by writing $\mathbf{z}$ and λ as $\mathbf{z} = \mathbf{z}_1 + i\mathbf{z}_2$ and $\lambda = \alpha + i\beta$, respectively. The reason for having an under-determined system of equations instead of a square system of equations is because, expanding $\mathbf{z}^H\mathbf{B}\mathbf{z} = 1$ gives only one real equation, since $\mathbf{B}$ is symmetric positive definite, while $(\mathbf{A} - \lambda\mathbf{B})\mathbf{z} = \mathbf{0}$ results in $2n$ real equations. This results in a real $(2n+1)$ under-determined system of nonlinear equations in $(2n+2)$ real unknowns. This will then be followed by presenting the real under-determined system of nonlinear equations and an explicit expression for its Jacobian.

Furthermore, we will show in the main result of this section-Theorem 4.2.1 that, if the eigenvalue of interest in $(\mathbf{A}, \mathbf{B})$ is algebraically simple, then the Jacobian has linearly independent rows at the root. We will find the right nullvector of the Jacobian at the root. We conclude the section by presenting Algorithm 14 for computing the complex eigenpair of the matrix pencil $(\mathbf{A}, \mathbf{B})$. A numerical example is given to illustrate the theory.

If we let $\mathbf{z} = \mathbf{z}_1 + i\mathbf{z}_2$ and $\lambda = \alpha + i\beta$, then the nonlinear system of equations (4.4) can be written as

$$\begin{aligned}(\mathbf{A} - \lambda\mathbf{B})\mathbf{z} &= \big[\mathbf{A} - (\alpha + i\beta)\mathbf{B}\big](\mathbf{z}_1 + i\mathbf{z}_2) \\ &= (\mathbf{A} - \alpha\mathbf{B})\mathbf{z}_1 + \beta\mathbf{B}\mathbf{z}_2 + i[(\mathbf{A} - \alpha\mathbf{B})\mathbf{z}_2 - \beta\mathbf{B}\mathbf{z}_1], \qquad (4.6)\end{aligned}$$

and

$$\mathbf{z}^H\mathbf{B}\mathbf{z} = \mathbf{z}_1^T\mathbf{B}\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{B}\mathbf{z}_2. \qquad (4.7)$$

Hence, (4.3) implies that

$$-\frac{1}{2}\mathbf{z}^H\mathbf{B}\mathbf{z} + \frac{1}{2} = -\frac{1}{2}(\mathbf{z}_1^T\mathbf{B}\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{B}\mathbf{z}_2) + \frac{1}{2} = \mathbf{0}.$$

Since $(\mathbf{A} - \lambda\mathbf{B})\mathbf{z} = \mathbf{0}$, we equate the real and imaginary parts of (4.6) to zero

and obtain the $2n$ real equations

$$(\mathbf{A} - \alpha\mathbf{B})\mathbf{z}_1 + \beta\mathbf{B}\mathbf{z}_2 = \mathbf{0},$$

and

$$(\mathbf{A} - \alpha\mathbf{B})\mathbf{z}_2 - \beta\mathbf{B}\mathbf{z}_1 = \mathbf{0}.$$

This means, $\mathbf{F}(\mathbf{v})$ consists of the $2n$ real equations arising from (4.6) and one real equation $-\frac{1}{2}(\mathbf{z}_1^T\mathbf{B}\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{B}\mathbf{z}_2) + \frac{1}{2} = 0$;

$$\mathbf{F}(\mathbf{v}) = \begin{bmatrix} (\mathbf{A} - \alpha\mathbf{B})\mathbf{z}_1 + \beta\mathbf{B}\mathbf{z}_2 \\ -\beta\mathbf{B}\mathbf{z}_1 + (\mathbf{A} - \alpha\mathbf{B})\mathbf{z}_2 \\ -\frac{1}{2}(\mathbf{z}_1^T\mathbf{B}\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{B}\mathbf{z}_2) + \frac{1}{2} \end{bmatrix} = \mathbf{0}, \tag{4.8}$$

where $\mathbf{F} : \mathbb{R}^{(2n+2)} \rightarrow \mathbb{R}^{(2n+1)}$. The Jacobian, $\mathbf{F}_\mathbf{v}(\mathbf{v})$ of $\mathbf{F}(\mathbf{v})$ with $\mathbf{v} = [\mathbf{z}_1, \mathbf{z}_2, \alpha.\beta]^T$ has the following explicit expression

$$\mathbf{F}_\mathbf{v}(\mathbf{v}) = \begin{bmatrix} (\mathbf{A} - \alpha\mathbf{B}) & \beta\mathbf{B} & -\mathbf{B}\mathbf{z}_1 & \mathbf{B}\mathbf{z}_2 \\ -\beta\mathbf{B} & (\mathbf{A} - \alpha\mathbf{B}) & -\mathbf{B}\mathbf{z}_2 & -\mathbf{B}\mathbf{z}_1 \\ -(\mathbf{B}\mathbf{z}_1)^T & -(\mathbf{B}\mathbf{z}_2)^T & 0 & 0 \end{bmatrix}, \tag{4.9}$$

and is a $(2n+1)$ by $(2n+2)$ real matrix. We define the real $2n$ by $2n$ matrix $\mathbf{M}$ as

$$\mathbf{M} = \begin{bmatrix} (\mathbf{A} - \alpha\mathbf{B}) & \beta\mathbf{B} \\ -\beta\mathbf{B} & (\mathbf{A} - \alpha\mathbf{B}) \end{bmatrix}. \tag{4.10}$$

Also, we form the $2n$ by 2 real matrix

$$\mathbf{N} = \begin{bmatrix} -\mathbf{B}\mathbf{z}_1 & \mathbf{B}\mathbf{z}_2 \\ -\mathbf{B}\mathbf{z}_2 & -\mathbf{B}\mathbf{z}_1 \end{bmatrix} = \begin{bmatrix} -\mathbf{B}_2\mathbf{w} & \mathbf{B}_2\mathbf{w}_1 \end{bmatrix}, \tag{4.11}$$

consisting of the product of $\mathbf{B}_2 = \begin{bmatrix} \mathbf{B} & \mathbf{O} \\ \mathbf{O} & \mathbf{B} \end{bmatrix}$ and the matrix of right nullvectors

(given in the next equation) of $\mathbf{M}$ at the root, where

$$\mathbf{w} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix}, \qquad \mathbf{w}_1 = \begin{bmatrix} \mathbf{z}_2 \\ -\mathbf{z}_1 \end{bmatrix}, \tag{4.12}$$

and $\mathbf{O}$ is the n by n zero matrix. The Jacobian (4.9) can be rewritten in the following partitioned form

$$\mathbf{F}_{\mathbf{v}}(\mathbf{v}) = \begin{bmatrix} \mathbf{M} & -\mathbf{B}_2\mathbf{w} & \mathbf{B}_2\mathbf{w}_1 \\ -(\mathbf{B}_2\mathbf{w})^T & 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{M} & \mathbf{N} \\ -(\mathbf{B}_2\mathbf{w})^T & \mathbf{0}^T \end{bmatrix}, \tag{4.13}$$

with $\mathbf{M}$, $\mathbf{N}$ defined in (4.10) and (4.11) respectively. Note that because at the root,

$$\begin{bmatrix} (\mathbf{A}-\alpha\mathbf{B}) & \beta\mathbf{B} \\ -\beta\mathbf{B} & (\mathbf{A}-\alpha\mathbf{B}) \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} = \begin{bmatrix} (\mathbf{A}-\alpha\mathbf{B})\mathbf{z}_1 + \beta\mathbf{B}\mathbf{z}_2 \\ (\mathbf{A}-\alpha\mathbf{B})\mathbf{z}_2 - \beta\mathbf{B}\mathbf{z}_1 \end{bmatrix} = \mathbf{0},$$

this implies that $\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix}$ or its nonzero scalar multiple is a right nullvector of $\mathbf{M}$. In the same vein, we find

$$\begin{bmatrix} (\mathbf{A}-\alpha\mathbf{B}) & \beta\mathbf{B} \\ -\beta\mathbf{B} & (\mathbf{A}-\alpha\mathbf{B}) \end{bmatrix} \begin{bmatrix} \mathbf{z}_2 \\ -\mathbf{z}_1 \end{bmatrix} = \begin{bmatrix} (\mathbf{A}-\alpha\mathbf{B})\mathbf{z}_2 - \beta\mathbf{B}\mathbf{z}_1 \\ -\{(\mathbf{A}-\alpha\mathbf{B})\mathbf{z}_1 + \beta\mathbf{B}\mathbf{z}_2\} \end{bmatrix} = \mathbf{0},$$

and $\begin{bmatrix} \mathbf{z}_2 \\ -\mathbf{z}_1 \end{bmatrix}$ or its nonzero scalar multiple is also a right nullvector of $\mathbf{M}$ at the root.

Since the eigenvalue λ of $(\mathbf{A}, \mathbf{B})$ is algebraically simple by assumption, then by (4.2), we need to give explicit expressions for the left nullvector of $(\mathbf{A} - \lambda\mathbf{B})$ in order to prove that the Jacobian has full row rank at the root. Observe that for all $\psi \in \mathcal{N}(\mathbf{A}-\lambda\mathbf{B})^H \backslash \{\mathbf{0}\}$, we define $\psi = \psi_1 + i\psi_2$, where $\psi_1, \psi_2 \in \mathbb{R}^n$, then this implies

$$\begin{aligned} \psi^H(\mathbf{A}-\lambda\mathbf{B}) &= (\psi_1^T - i\psi_2^T)[(\mathbf{A}-\alpha\mathbf{B}) - i\beta\mathbf{B}] \\ &= \psi_1^T(\mathbf{A}-\alpha\mathbf{B}) - \beta\psi_2^T\mathbf{B} - i[\beta\psi_1^T\mathbf{B} + \psi_2^T(\mathbf{A}-\alpha\mathbf{B})] = \mathbf{0}^T. \end{aligned}$$

Hence, $\psi_1^T(\mathbf{A}-\alpha\mathbf{B}) - \beta\psi_2^T\mathbf{B} = \mathbf{0}^T$ and $\beta\psi_1^T\mathbf{B} + \psi_2^T(\mathbf{A}-\alpha\mathbf{B}) = \mathbf{0}^T$. The impli-

cation of this is that

$$[\psi_1^T \quad \psi_2^T]\mathbf{M} = [\psi_1^T \quad \psi_2^T]\begin{bmatrix}(\mathbf{A}-\alpha\mathbf{B}) & \beta\mathbf{B} \\ -\beta\mathbf{B} & (\mathbf{A}-\alpha\mathbf{B})\end{bmatrix}$$
$$= [\psi_1^T(\mathbf{A}-\alpha\mathbf{B}) - \beta\psi_2^T\mathbf{B} \qquad \beta\psi_1^T\mathbf{B} + \psi_2^T(\mathbf{A}-\alpha\mathbf{B})] = \mathbf{0}^T,$$

which means, $[\psi_1^T,\ \psi_2^T]$ or its nonzero scalar multiple is a left nullvector of $\mathbf{M}$. Similarly,

$$[\psi_2^T \quad -\psi_1^T]\mathbf{M} = [\psi_2^T \quad -\psi_1^T]\begin{bmatrix}(\mathbf{A}-\alpha\mathbf{B}) & \beta\mathbf{B} \\ -\beta\mathbf{B} & (\mathbf{A}-\alpha\mathbf{B})\end{bmatrix}$$
$$= [\beta\psi_1^T\mathbf{B} + \psi_2^T(\mathbf{A}-\alpha\mathbf{B}) \qquad -\{\psi_1^T(\mathbf{A}-\alpha\mathbf{B}) - \beta\psi_2^T\mathbf{B}\}] = \mathbf{0}^T,$$

and it shows that $[\psi_2^T,\ -\psi_1^T]$ is also a left nullvector of $\mathbf{M}$.

So we form the matrix $\mathbf{C}$ consisting of the 2-dimensional left nullvectors of $\mathbf{M}$ at the root (in practice $\mathbf{C}$ is not computed), as

$$\mathbf{C} = \begin{bmatrix}\psi_1 & \psi_2 \\ \psi_2 & -\psi_1\end{bmatrix}. \tag{4.14}$$

Now, observe that the condition (4.2), implies

$$\psi^H\mathbf{B}\mathbf{z} = [\psi_1^T\mathbf{B}\mathbf{z}_1 + \psi_2^T\mathbf{B}\mathbf{z}_2] + i[\psi_1^T\mathbf{B}\mathbf{z}_2 - \psi_2^T\mathbf{B}\mathbf{z}_1] \neq 0.$$

Since $\psi^H\mathbf{B}\mathbf{z} \neq 0$, this implies

$$[\psi_1^T\mathbf{B}\mathbf{z}_1 + \psi_2^T\mathbf{B}\mathbf{z}_2]^2 + [\psi_1^T\mathbf{B}\mathbf{z}_2 - \psi_2^T\mathbf{B}\mathbf{z}_1]^2 \neq 0. \tag{4.15}$$

Before we continue with the rest of the analysis, we present the main result of this section which shows that the Jacobian (4.9) has a one dimensional nullvector at the root.

Theorem 4.2.1. *Assume that the eigenpair $(\mathbf{z}, \lambda)$ of the pencil $(\mathbf{A}, \mathbf{B})$ is algebraically simple. If $\mathbf{z}_1$ and $\mathbf{z}_2$ are nonzero vectors, then $\phi = \{\tau[\mathbf{z}_2^T, -\mathbf{z}_1^T, 0, 0], \tau \in \mathbb{R}\}$ is the eigenspace corresponding to the zero eigenvalue of $\mathbf{F}_\mathbf{v}(\mathbf{v})$ at the root.*

Proof: Post-multiply $\mathbf{F}_\mathbf{v}(\mathbf{v})$ by the unknown nonzero vector $\phi = [\mathbf{p}', \mathbf{q}']^T$,

equate to the zero vector and solve

$$\begin{bmatrix} \mathbf{M} & \mathbf{N} \\ -(\mathbf{B}_2\mathbf{w})^T & \mathbf{0}^T \end{bmatrix} \begin{bmatrix} \mathbf{p}' \\ \mathbf{q}' \end{bmatrix} = \mathbf{0},$$

where $\mathbf{M}$ and $\mathbf{N}$ are as defined in (4.10) and (4.11) respectively. After expanding, we have the following set of equations

$$\mathbf{M}\mathbf{p}' + \mathbf{N}\mathbf{q}' = \mathbf{0} \tag{4.16}$$

$$\mathbf{w}^T\mathbf{B}_2\mathbf{p}' = 0. \tag{4.17}$$

Let $\mathbf{H} = \mathbf{C}^T\mathbf{N}$, for all $\mathbf{C} \in \mathcal{N}(\mathbf{M}^T)\backslash\{\mathbf{0}\}$ as in (4.14). This means,

$$\begin{aligned} \mathbf{H} &= \begin{bmatrix} \boldsymbol{\psi}_1^T & \boldsymbol{\psi}_2^T \\ \boldsymbol{\psi}_2^T & -\boldsymbol{\psi}_1^T \end{bmatrix} \begin{bmatrix} \mathbf{B} & \\ & \mathbf{B} \end{bmatrix} \begin{bmatrix} -\mathbf{z}_1 & \mathbf{z}_2 \\ -\mathbf{z}_2 & -\mathbf{z}_1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\psi}_1^T & \boldsymbol{\psi}_2^T \\ \boldsymbol{\psi}_2^T & -\boldsymbol{\psi}_1^T \end{bmatrix} \begin{bmatrix} -\mathbf{B}\mathbf{z}_1 & \mathbf{B}\mathbf{z}_2 \\ -\mathbf{B}\mathbf{z}_2 & -\mathbf{B}\mathbf{z}_1 \end{bmatrix} \\ &= \begin{bmatrix} -(\boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_1 + \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_2) & \boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_2 - \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_1 \\ \boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_2 - \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_1 & (\boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_1 + \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_2) \end{bmatrix}. \end{aligned}$$

By premultiplying both sides of (4.16) by $\mathbf{C}^T$, we obtain

$$\mathbf{C}^T\mathbf{M}\mathbf{p}' + \mathbf{C}^T\mathbf{N}\mathbf{q}' = \mathbf{0}. \tag{4.18}$$

But, $\mathbf{C}^T\mathbf{M} = \mathbf{0}^T$. Consequently, we are left with $\mathbf{C}^T\mathbf{N}\mathbf{q}' = \mathbf{0}$, or

$$\mathbf{H}\mathbf{q}' = \mathbf{C}^T\mathbf{N}\mathbf{q}' = \begin{bmatrix} -(\boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_1 + \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_2) & \boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_2 - \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_1 \\ \boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_2 - \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_1 & (\boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_1 + \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_2) \end{bmatrix} \mathbf{q}' = \mathbf{0}.$$

Now,

$$\det \mathbf{H} = -\{(\boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_1 + \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_2)^2 + (\boldsymbol{\psi}_1^T\mathbf{B}\mathbf{z}_2 - \boldsymbol{\psi}_2^T\mathbf{B}\mathbf{z}_1)^2\} \neq 0,$$

using (4.15), which implies $\mathbf{H}$ is nonsingular. Thus, $\mathbf{q}' = \mathbf{0}$. Equation (4.16) now becomes $\mathbf{M}\mathbf{p}' = \mathbf{0}$, meaning that $\mathbf{p}' \in \mathcal{N}(\mathbf{M})$, $\mathbf{p}' = \mu\mathbf{w} + \tau\mathbf{w}_1$. From (4.17),

$$0 = \mathbf{w}^T\mathbf{B}_2\mathbf{p}' = \mu\mathbf{w}^T\mathbf{B}_2\mathbf{w} + \tau\mathbf{w}^T\mathbf{B}_2\mathbf{w}_1.$$

Now, because $\mathbf{w}^T\mathbf{B}_2\mathbf{w}_1 = 0$ and $\mathbf{w}^T\mathbf{B}_2\mathbf{w} \neq 0$, we have $\mu = 0$ and so

$$\mathbf{p}' = \tau\mathbf{w}_1.$$

Hence, for all $\tau \in \mathbb{R}\backslash\{0\}$, $\mathbf{p}' = [\tau\mathbf{z}_2, -\tau\mathbf{z}_1]^T \in \mathcal{N}(\mathbf{M})$ also satisfies equation (4.17). Therefore, we obtain $\phi = \tau[\mathbf{z}_2, -\mathbf{z}_1, 0, 0]^T$ as the only nonzero nullvector of $\mathbf{F}_\mathbf{v}(\mathbf{v})$. ∎

The next result is a corollary to Theorem 4.2.1 and it shows that the Jacobian (4.9) has linearly independent rows at the root.

Corollary 4.2.1. *If the eigenpair $(\mathbf{z}, \lambda)$ of $(\mathbf{A}, \mathbf{B})$ is algebraically simple, then the Jacobian $\mathbf{F}_\mathbf{v}(\mathbf{v})$ in* (4.13) *is of full rank at the root.*

Proof: Since Theorem 4.2.1 guarantees the existence of a single nonzero nullvector of $\mathbf{F}_\mathbf{v}(\mathbf{v})$ at the root, then rank $\big(\mathbf{F}_\mathbf{v}(\mathbf{v})\big) = 2n + 1$ (using the dimension theorem, see, for example, [39]). Therefore, the Jacobian (4.9) is of full rank at the root. ∎

Next, in order to solve the under-determined system of nonlinear equations (4.8), we need to linearize $\mathbf{F}(\mathbf{v}) = \mathbf{0}$. After linearizing $\mathbf{F}(\mathbf{v}) = \mathbf{0}$, we have to solve the following under-determined linear system of equations

$$\mathbf{F}_\mathbf{v}(\mathbf{v}^{(k)})\Delta\mathbf{v}^{(k)} = -\mathbf{F}(\mathbf{v}^{(k)}). \tag{4.19}$$

Hence, solving for $\Delta\mathbf{v}^{(k)}$ in $\mathbf{F}_\mathbf{v}(\mathbf{v}^{(k)})\Delta\mathbf{v}^{(k)} = -\mathbf{F}(\mathbf{v}^{(k)})$, involves solving a $2n+1$ real under-determined linear system of equations for the $2n+2$ real unknowns $\Delta\mathbf{v}^{(k)} = [\Delta\mathbf{z}_1^{(k)}, \Delta\mathbf{z}_2^{(k)}, \Delta\alpha^{(k)}, \Delta\beta^{(k)}]^T$. Following the discussion in Section 1.5.2, we find the reduced QR factorization $\mathbf{F}_\mathbf{v}(\mathbf{v}^{(k)})^T = \mathbf{QR}$, where in this case $\mathbf{Q}$ and $\mathbf{R}$ are $(2n+2)$ by $(2n+1)$ and $(2n+1)$ by $(2n+1)$ real matrices respectively. Hence, we solve $\mathbf{R}^T\mathbf{g}^{(k)} = -\mathbf{F}(\mathbf{v}^{(k)})$ for $\mathbf{g}^{(k)}$ and then obtain the solution to (4.19) as

$$\Delta\mathbf{v}^{(k)} = \mathbf{Q}\mathbf{g}^{(k)},$$

and update $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta\mathbf{v}^{(k)}$. Since we have shown that the Jacobian has linearly independent rows in Theorem 4.2.1, the whole analysis now gives rise to Algorithm 14, namely, the **Gauss-Newton method** applied to $\mathbf{F}(\mathbf{v}) = \mathbf{0}$.

Algorithm 14 Eigenpair Computation using Gauss-Newton's method

Input: $\mathbf{A}, \mathbf{B}$, $\mathbf{v}^{(0)} = [\mathbf{z}_1^{(0)}, \mathbf{z}_2^{(0)}, \alpha^{(0)}, \beta^{(0)}]^T$, $k_{\max}$ and tol.

1: **for** $k = 0, 1, 2, \ldots,$ until convergence **do**
2: Find the reduced QR factorisation of $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})^T = \mathbf{QR}$.
3: Solve $\mathbf{R}^T\mathbf{g}^{(k)} = -\mathbf{F}(\mathbf{v}^{(k)})$ for $\mathbf{g}^{(k)}$ in (4.9).
4: Compute $\Delta\mathbf{v}^{(k)} = \mathbf{Q}\mathbf{g}^{(k)}$ for $\Delta\mathbf{v}^{(k)}$ using (4.8).
5: Update $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta\mathbf{v}^{(k)}$.
6: **end for**

Output: $\mathbf{v}^{(k_{\max})}$.

The stopping condition for Algorithm 14 is

$$\|\Delta\mathbf{v}^{(k)}\| \leq tol.$$

Next, we give the following numerical example to illustrate the above theory.

Example 4.2.1. *Consider the 200 by 200 matrix* $\mathbf{A}$ `bwm200.mtx` *from the matrix market library [9]. It is the discretised Jacobian of the Brusselator wave model for a chemical reaction. The resulting eigenvalue problem with* $\mathbf{B} = \mathbf{I}$ *was also studied in [48] and we are interested in finding the rightmost eigenvalue of* $\mathbf{A}$ *which is closest to the imaginary axis and its corresponding eigenvector.*

In this example, we take $\alpha^{(0)} = 0.0, \beta^{(0)} = 2.5$ *in line with [48] and took* $\mathbf{z}_1^{(0)} = \mathbf{1}/2\|\mathbf{1}\|$ *and* $\mathbf{z}_2^{(0)} = \frac{\sqrt{3}}{2}\mathbf{1}/\|\mathbf{1}\|$, *where* $\mathbf{1}$ *is the vector of all ones. Algorithm 14 is stopped as soon as* $\|\Delta\mathbf{v}^{(k)}\|$ *is less than or equal to* 5.6×10^{-14}. *The computed eigenpairs are shown in Table 4.1. Observe that we obtained quadratic convergence*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\|\mathbf{w}^{(k+1)} - \mathbf{w}^{(k)}\|$	$\|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\|$	$\|\Delta\mathbf{v}^{(k)}\|$	$\|\mathbf{F}(\mathbf{v}^{(k)})\|$
0	0.00000e+00	2.50000	3.8e+00	7.8e-01	3.9e+00	3.6e+01
1	2.34253e-01	1.75371	1.8e+00	2.2e-01	1.8e+00	7.8e+00
2	1.18745e-01	1.94460	8.1e-01	1.4e-01	8.2e-01	1.7e+00
3	4.47044e-02	2.06484	2.5e-01	7.0e-02	2.6e-01	3.4e-01
4	8.82702e-03	2.12479	3.1e-02	1.7e-02	3.5e-02	3.7e-02
5	2.48114e-04	2.13905	4.8e-04	5.2e-04	7.1e-04	7.1e-04
6	1.80714e-05	2.13950	1.2e-07	2.5e-07	2.8e-07	2.8e-07
7	1.81999e-05	2.13950	2.1e-14	2.9e-14	3.6e-14	6.0e-14

Table 4.1: Values of $\alpha^{(k)}$ and $\beta^{(k)}$ of Example 4.2.1. Columns 6 and 7 show that the results converged quadratically for $k = 3, 4, 5, 6$ and 7.

from the second to the last and the last columns of Table 4.1 for $k = 3, 4, 5, 6$ *and 7.*

At the root, the condition number of $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})$ *is approximately* 3×10^3. $\mathbf{w}^{(k)}$ *in the above table represents* $[{\mathbf{z}_1^{(k)}}^T, {\mathbf{z}_2^{(k)}}^T]$ *and* $\boldsymbol{\lambda}^{(k)} = [\alpha^{(k)}, \beta^{(k)}]$.

Next, we show that the solution $\Delta\mathbf{v}^{(k)}$ obtained by solving the under-determined system of nonlinear equations (4.19) is equivalent to those obtained by solving a square, augmented linear system.

Lemma 4.2.1. *Let* $\boldsymbol{n}^{(k)}$ *be the exact nullvector of* $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})$. *The solution* $\Delta\mathbf{v}^{(k)}$ *can be obtained via:*

(a). solving the under-determined linear system of $(2n+1)$ *real equations for the* $(2n+2)$ *real unknowns* $\Delta\mathbf{v}^{(k)}$ (4.19) *and updating* $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta\mathbf{v}^{(k)}$, *or*

(b). solving the square linear system of $(2n+2)$ *real equations* (4.39) *and updating* $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta\mathbf{v}^{(k)}$.

(Here, we neglect round off errors).

Proof: Assume by contradiction that $\Delta\mathbf{v}^{(k)} \neq \Delta\mathbf{v}_1^{(k)}$, where

$$\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})\Delta\mathbf{v}^{(k)} = -\mathbf{F}(\mathbf{v}^{(k)}),$$

and

$$\begin{bmatrix} \mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)}) \\ {\boldsymbol{n}^{(k)}}^T \end{bmatrix} \Delta\mathbf{v}_1^{(k)} = - \begin{bmatrix} \mathbf{F}(\mathbf{v}^{(k)}) \\ 0 \end{bmatrix}. \tag{4.20}$$

Since $\boldsymbol{n}^{(k)}$ is an exact nullvector of $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})$ by definition, $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})\boldsymbol{n}^{(k)} = \mathbf{0}$. From Lemma 1.5.1,

$${\boldsymbol{n}^{(k)}}^T\Delta\mathbf{v}^{(k)} = 0. \tag{4.21}$$

Now, by subtracting $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})\Delta\mathbf{v}_1^{(k)} = -\mathbf{F}(\mathbf{v}^{(k)})$ from $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})\Delta\mathbf{v}^{(k)} = -\mathbf{F}(\mathbf{v}^{(k)})$, results in $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})\left(\Delta\mathbf{v}^{(k)} - \Delta\mathbf{v}_1^{(k)}\right) = \mathbf{0}$. Which implies $\Delta\mathbf{v}^{(k)} - \Delta\mathbf{v}_1^{(k)} = \tau\boldsymbol{n}^{(k)}$. After taking the inner product of both sides with $\boldsymbol{n}^{(k)}$, we obtain

$${\boldsymbol{n}^{(k)}}^T\Delta\mathbf{v}^{(k)} - {\boldsymbol{n}^{(k)}}^T\Delta\mathbf{v}_1^{(k)} = \tau\|\boldsymbol{n}^{(k)}\|^2.$$

The first term on the left hand side of the equation above is zero by virtue of (4.21) and the second term is also zero, from (4.20). Accordingly, $\tau\|\boldsymbol{n}^{(k)}\|^2 = 0$

and $\tau = 0$. Which means that $\Delta\mathbf{v}^{(k)} - \Delta\mathbf{v}_1^{(k)} = 0$ and $\Delta\mathbf{v}^{(k)} = \Delta\mathbf{v}_1^{(k)}$, contradicting the assumption that $\Delta\mathbf{v}^{(k)} \neq \Delta\mathbf{v}_1^{(k)}$. Consequently, $\Delta\mathbf{v}^{(k)} = \Delta\mathbf{v}_1^{(k)}$. ■

4.3 A Theoretical form for the Nullvector of the Jacobian (4.9)

In the proof of Lemma 4.2.1 at the tail end of last section, we made use of the exact nullvector (which we do not compute in practice) of the Jacobian (4.9). In this section, we give a theoretical expression for the exact nullvector of the Jacobian (4.9) when not at the root. To do this, we rewrite the under-determined linear system of equations (4.19) in a compressed form, present two important theoretical relationships: (4.27) and (4.28) for the exact nullvector of the Jacobian. These expressions will be used extensively in Sections 4.4 and 4.6 ahead.

Note that the matrix $\mathbf{M}$ defined by (4.10) is singular at the root. However, this section is anchored on the assumption that when $\mathbf{v}$ is not at the root, $\mathbf{M}$ is nonsingular.

First, we define the $2n$ by $2n$ matrix $\mathbf{J}$ as (see, for example [22])

$$\mathbf{J} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{I} & 0 \end{bmatrix}, \tag{4.22}$$

and note that

$$\mathbf{J}\mathbf{w} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{I} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{z}_2 \\ -\mathbf{z}_1 \end{bmatrix} = \mathbf{w}_1, \tag{4.23}$$

defined by (4.12).

The matrix $\mathbf{J}$ satisfies the following properties:

1. $\mathbf{J}^T = -\mathbf{J}$.
2. $\mathbf{J}^T\mathbf{J} = \mathbf{I}_{2n}$, where $\mathbf{I}_{2n}$ is the $2n$ by $2n$ identity matrix.
3. $\mathbf{J}^2 = -\mathbf{I}_{2n}$.

4. $\mathbf{J}$ commutes with $\mathbf{M}$ and $\mathbf{B}_2$, *i.e.*, $\mathbf{JM} = \mathbf{MJ}$ and $\mathbf{JB}_2 = \mathbf{B}_2\mathbf{J}$.

5. For $\mathbf{w} \in \mathbb{R}^{2n}$, $\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{w} = \mathbf{w}^T\mathbf{J}\mathbf{B}_2\mathbf{w} = 0$.

6. Let $\mathbf{u}$ be an unknown vector that solves $\mathbf{Mu} = \mathbf{B}_2\mathbf{w}$. By premultiplying both sides by $\mathbf{J}$ we obtain $\mathbf{JMu} = \mathbf{JB}_2\mathbf{w}$ and hence $\mathbf{MJu} = \mathbf{JB}_2\mathbf{w}$ by the commutativity of $\mathbf{M}$ and $\mathbf{J}$. Therefore,

$$\mathbf{Mu} = \mathbf{B}_2\mathbf{w}, \quad \text{implies} \quad \mathbf{M}(\mathbf{Ju}) = \mathbf{JB}_2\mathbf{w}. \tag{4.24}$$

The equation $\mathbf{Mu} = \mathbf{B}_2\mathbf{w}$ stems from expanding the shifted system $(\mathbf{A} - \sigma\mathbf{B})\mathbf{y} = \mathbf{Bz}$, into its real and imaginary parts as in [48] for $\sigma = \alpha + i\beta$ and $\mathbf{z} = \mathbf{z}_1 + i\mathbf{z}_2$. For ease of notation and for the rest of this chapter, we shall drop the superscripts $^{(k)}$ and write $\mathbf{w}^+ = \mathbf{w} + \Delta\mathbf{w}$ where $\mathbf{w}^+ = \mathbf{w}^{(k+1)}$, replace $\mathbf{w}^{(k)}$ and $[{\Delta\mathbf{z}_1^{(k)}}^T, {\Delta\mathbf{z}_2^{(k)}}^T]$ with $\mathbf{w}$ and $\Delta\mathbf{w}$ respectively *e.t.c.*

As earlier stated, we assume that the $2n$ by $2n$ matrix $\mathbf{M}$ is nonsingular except at the root. For the rest of this section, our aim is to give an explicit theoretical expression for the nullvector of (4.9).

Let the exact nullvector $\boldsymbol{n}$ of

$$\mathbf{F}_{\mathbf{v}}(\mathbf{v}) = \begin{bmatrix} \mathbf{M} & -\mathbf{B}_2\mathbf{w} & \mathbf{B}_2\mathbf{J}\mathbf{w} \\ -(\mathbf{B}_2\mathbf{w})^T & 0 & 0 \end{bmatrix},$$

be defined as $\boldsymbol{n} = [\boldsymbol{n}_{\mathbf{w}}^T, \boldsymbol{n}_\alpha, \boldsymbol{n}_\beta]$, where $\boldsymbol{n}_{\mathbf{w}} \in \mathbb{R}^{2n}$, $\boldsymbol{n}_\alpha$ and $\boldsymbol{n}_\beta$ are real scalars, $\mathbf{Jw}$ and $\mathbf{M}$ are defined respectively by (4.23) and (4.10). Hence,

$$\begin{bmatrix} \mathbf{M} & -\mathbf{B}_2\mathbf{w} & \mathbf{B}_2\mathbf{J}\mathbf{w} \\ -(\mathbf{B}_2\mathbf{w})^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{n}_{\mathbf{w}} \\ \boldsymbol{n}_\alpha \\ \boldsymbol{n}_\beta \end{bmatrix} = \mathbf{0},$$

then after expanding the matrix-vector multiplication, we obtain

$$\mathbf{M}\boldsymbol{n}_{\mathbf{w}} - \boldsymbol{n}_\alpha\mathbf{B}_2\mathbf{w} + \boldsymbol{n}_\beta(\mathbf{B}_2\mathbf{J}\mathbf{w}) = \mathbf{0} \tag{4.25}$$

$$\mathbf{w}^T\mathbf{B}_2\boldsymbol{n}_{\mathbf{w}} = 0. \tag{4.26}$$

From (4.25), $\mathbf{M}\boldsymbol{n}_{\mathbf{w}} = \boldsymbol{n}_\alpha\mathbf{B}_2\mathbf{w} - \boldsymbol{n}_\beta(\mathbf{B}_2\mathbf{J}\mathbf{w})$, using the fact that $\mathbf{J}$ commutes with

$\mathbf{B}_2$ and $\mathbf{M}$, and using (4.24) with $\mathbf{B}_2 = \mathbf{I}_{2n}$ we obtain

$$\boldsymbol{n}_{\mathbf{w}} = \boldsymbol{n}_{\alpha}\mathbf{u} - \boldsymbol{n}_{\beta}\mathbf{J}\mathbf{u}. \tag{4.27}$$

Since $\mathbf{w}$ is $\mathbf{B}_2$-orthogonal to $\boldsymbol{n}_{\mathbf{w}}$ by virtue of (4.26), taking the $\mathbf{B}_2$-inner product of both sides of the above with $\mathbf{w}$ yields

$$\mathbf{w}^T\mathbf{B}_2\boldsymbol{n}_{\mathbf{w}} = \boldsymbol{n}_{\alpha}(\mathbf{w}^T\mathbf{B}_2\mathbf{u}) - \boldsymbol{n}_{\beta}(\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u}) = \mathbf{0}.$$

We may choose

$$\boldsymbol{n}_{\alpha} = \mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u}, \quad \text{and} \quad \boldsymbol{n}_{\beta} = \mathbf{w}^T\mathbf{B}_2\mathbf{u}, \tag{4.28}$$

since we never normalise $\boldsymbol{n}$. Hence, $\boldsymbol{n}_{\mathbf{w}}$ is given by (4.27) with $\boldsymbol{n}_{\alpha}$ and $\boldsymbol{n}_{\beta}$ by (4.28). So we have a formula for $\boldsymbol{n}_{\mathbf{w}}$ in terms of $\mathbf{w}$ and $\mathbf{u}$ obtained from (4.24). Therefore,

$$\boldsymbol{n} = [\boldsymbol{n}_{\mathbf{w}}^T, \boldsymbol{n}_{\alpha}, \boldsymbol{n}_{\beta}] = [(\boldsymbol{n}_{\alpha}\mathbf{u} - \boldsymbol{n}_{\beta}\mathbf{J}\mathbf{u})^T, (\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u}), (\mathbf{w}^T\mathbf{B}_2\mathbf{u})].$$

We emphasise that in practice, we would never compute the solution of (4.24). It will be used for purely theoretical purposes since we know that the Gauss-Newton solution, $\Delta\mathbf{v}$, is orthogonal to $\boldsymbol{n}$.

4.4 Square System of Equations for The Numerical Computation of the Complex Eigenvalues of a Matrix for B = I

In the preceding section, we presented two main important theoretical relationships, (4.27) and (4.28). In this section, we will make use of these relationships in our discussion but only in the special case in which $\mathbf{B} = \mathbf{I}$. Moreover, in Section (4.2), we saw that the solution to the under-determined system of nonlinear equations (4.8) for the numerical computation of the complex eigenpair $(\mathbf{z}, \lambda)$ of the pencil $(\mathbf{A}, \mathbf{B})$ can be solved by the Gauss-Newton method via QR factorization. It was also stated in Lemma 1.5.1 that the minimum norm solution to the resulting linear system of equations is orthogonal to the

nullspace. However, in Section 4.2, we used the result of Lemma 1.5.1 to add an extra equation to the under-determined linear system of equations, so as to obtain a square one. This is because, at each iteration of the computation, ${\boldsymbol{n}^{(k)}}^T \Delta\mathbf{v}^{(k)} = 0$ and so it does not change the solution, even though the square linear system of equations gives a unique solution because the augmented Jacobian is nonsingular.

Nevertheless, as mentioned in the last section, we would never compute $\boldsymbol{n}$ in practice, but Theorem 4.2.1 guarantees the existence of a unique nullvector $\boldsymbol{\phi}$ at the root. We will use $\boldsymbol{\phi}^{(k)}$ defined by $\boldsymbol{\phi}^{(k)} = [\mathbf{z}_2^{(k)}, -\mathbf{z}_1^{(k)}, 0, 0]$ as an approximation to the exact nullvector $\boldsymbol{n}$ and show that the solution obtained by solving (4.19) is equivalent to the solution obtained by solving

$$\begin{bmatrix} \mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)}) \\ {\boldsymbol{\phi}^{(k)}}^T \end{bmatrix} \Delta\mathbf{v}^{(k)} = - \begin{bmatrix} \mathbf{F}(\mathbf{v}^{(k)}) \\ 0 \end{bmatrix}, \tag{4.29}$$

in the absence of round off errors. To do this, we will show that ${\boldsymbol{\phi}^{(k)}}^T \Delta\mathbf{v}^{(k)} = 0$ for each k, where $\Delta\mathbf{v}^{(k)}$ is given by (4.19) and this is the key result in this section.

This section is structured as follows, we begin by adding the extra equation ${\boldsymbol{n}^{(k)}}^T \Delta\mathbf{v}^{(k)} = 0$ to (4.19) in order to obtain the square linear system of equations (4.20). The main result in this section is Theorem 4.4.1, and Algorithm 15 is presented for computing the algebraically simple eigenpair of $\mathbf{A}$. Note that since $\mathbf{M}$ has been shown to be singular at the root in section 4.2, this section is anchored on the assumption that when $\mathbf{v}$ is not at the root, $\mathbf{M}$ is nonsingular, but this is acceptable since we use the construction here to prove a theoretical result about the correction $\Delta\mathbf{v}^{(k)}$ while not at the root.

Consider the problem of solving the under-determined linear system of equations (4.19) for the $(2n+2)$ real unknowns $\Delta\mathbf{v} = [\Delta\mathbf{w}^T, \Delta\alpha, \Delta\beta]$. It was stated in Lemma 1.5.1 that the minimum norm solution to an under-determined linear system of equations is orthogonal to the nullspace. It is an application of this result that yields the following important relationship,

$$0 = \boldsymbol{n}^T \Delta\mathbf{v} = \boldsymbol{n}_{\mathbf{w}}^T \Delta\mathbf{w} + \boldsymbol{n}_\alpha \Delta\alpha + \boldsymbol{n}_\beta \Delta\beta, \tag{4.30}$$

where we have dropped the superscript $^{(k)}$ in $\alpha, \beta, \boldsymbol{n}, \mathbf{w}$ and $\mathbf{v}$. We begin by writing the linear system of equations (4.19) in expanded form as

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{w} \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} -\mathbf{M}\mathbf{w} \\ \frac{1}{2}(\mathbf{w}^T\mathbf{w} - 1) \end{bmatrix}, \tag{4.31}$$

or,

$$\begin{aligned} \mathbf{M}\Delta\mathbf{w} - \Delta\alpha\mathbf{w} + \Delta\beta\mathbf{J}\mathbf{w} &= -\mathbf{M}\mathbf{w} \\ -\mathbf{w}^T\Delta\mathbf{w} &= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \frac{1}{2}. \end{aligned}$$

After rearrangement, the first equation reduces to

$$\mathbf{M}\mathbf{w}^+ - \Delta\alpha\mathbf{w} + \Delta\beta\mathbf{J}\mathbf{w} = \mathbf{0}. \tag{4.32}$$

By multiplying both sides of the second equation by 2, we obtain:

$$2\mathbf{w}^T\Delta\mathbf{w} + \mathbf{w}^T\mathbf{w} = 1.$$

This in turn reduces to

$$\mathbf{w}^T(\mathbf{w} + 2\Delta\mathbf{w}) = 1. \tag{4.33}$$

Since $\mathbf{w}^+ = \mathbf{w} + \Delta\mathbf{w}$, $2\Delta\mathbf{w} = 2\mathbf{w}^+ - 2\mathbf{w}$ and $\mathbf{w} + 2\Delta\mathbf{w} = 2\mathbf{w}^+ - \mathbf{w}$, then $\mathbf{w}^T(\mathbf{w} + 2\Delta\mathbf{w}) = \mathbf{w}^T(2\mathbf{w}^+ - \mathbf{w}) = 2\mathbf{w}^T\mathbf{w}^+ - \mathbf{w}^T\mathbf{w}$. Consequently,

$$\mathbf{w}^T\mathbf{w}^+ = \frac{1}{2}(\mathbf{w}^T\mathbf{w} + 1). \tag{4.34}$$

The combined set of equations (4.32) and (4.34), which is the simplified form of (4.31), can be expressed as:

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w}^+ \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ -\frac{1}{2}(\mathbf{w}^T\mathbf{w} + 1) \end{bmatrix}. \tag{4.35}$$

Now, if we expand along the first row of (4.35), then

$$\mathbf{M}\mathbf{w}^{+} = \Delta\alpha\mathbf{w} - \Delta\beta\mathbf{J}\mathbf{w}. \tag{4.36}$$

This means that we could solve (4.35) by solving

$$\mathbf{M}\mathbf{u} = \mathbf{w}, \quad \text{and} \quad \mathbf{M}\mathbf{J}\mathbf{u} = \mathbf{J}\mathbf{w}, \tag{4.37}$$

(by Property 6 of $\mathbf{J}$ after (4.23)), for $\mathbf{u}$, after which the solution of (4.36) is given by

$$\mathbf{w}^{+} = \Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}. \tag{4.38}$$

If we add the nullvector $\boldsymbol{n}$ to the last row of (4.31) with $\mathbf{B} = \mathbf{I}$ and using (4.30), then

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^{T} & 0 & 0 \\ \boldsymbol{n}_{\mathbf{w}}^{T} & \boldsymbol{n}_{\alpha} & \boldsymbol{n}_{\beta} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{w} \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} -\mathbf{M}\mathbf{w} \\ \frac{1}{2}(\mathbf{w}^{T}\mathbf{w} - 1) \\ 0 \end{bmatrix}. \tag{4.39}$$

One can also add $\boldsymbol{n}$ to the last row of (4.35) to yield

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^{T} & 0 & 0 \\ \boldsymbol{n}_{\mathbf{w}}^{T} & \boldsymbol{n}_{\alpha} & \boldsymbol{n}_{\beta} \end{bmatrix} \begin{bmatrix} \mathbf{w}^{+} \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ -\frac{1}{2}(\mathbf{w}^{T}\mathbf{w} + 1) \\ \boldsymbol{n}_{\mathbf{w}}^{T}\mathbf{w} \end{bmatrix}. \tag{4.40}$$

By expanding the middle row of (4.40), $\mathbf{w}^{T}\mathbf{w}^{+} = \frac{1}{2}(\mathbf{w}^{T}\mathbf{w} + 1)$. But from (4.38), $\mathbf{w}^{+} = \Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}$. This implies that, by taking the inner product of both sides with $\mathbf{w}$, yields

$$\mathbf{w}^{T}\mathbf{w}^{+} = \Delta\alpha(\mathbf{w}^{T}\mathbf{u}) - \Delta\beta(\mathbf{w}^{T}\mathbf{J}\mathbf{u}) = \frac{1}{2}(\mathbf{w}^{T}\mathbf{w} + 1).$$

Using the definition (4.28) for $\boldsymbol{n}_{\alpha}$ and $\boldsymbol{n}_{\beta}$ with $\mathbf{B} = \mathbf{I}$, we obtain

$$\boldsymbol{n}_{\beta}\Delta\alpha - \boldsymbol{n}_{\alpha}\Delta\beta = \frac{1}{2}(\mathbf{w}^{T}\mathbf{w} + 1), \tag{4.41}$$

where the unknown quantities $\Delta\alpha$ and $\Delta\beta$ are to be determined, so we need an extra equation to be able to do so. Note that by using $\boldsymbol{n}_{\mathbf{w}} = \boldsymbol{n}_{\alpha}\mathbf{u} - \boldsymbol{n}_{\beta}\mathbf{J}\mathbf{u}$, and

(4.28) we can simplify

$$\begin{aligned}
\boldsymbol{n}_{\mathbf{w}}^T\mathbf{w} &= \boldsymbol{n}_\alpha \mathbf{u}^T\mathbf{w} - \boldsymbol{n}_\beta \mathbf{u}^T\mathbf{J}^T\mathbf{w} \\
&= \boldsymbol{n}_\alpha \mathbf{u}^T\mathbf{w} + \boldsymbol{n}_\beta \mathbf{u}^T\mathbf{J}\mathbf{w} \\
&= (\mathbf{w}^T\mathbf{J}\mathbf{u})(\mathbf{u}^T\mathbf{w}) + (\mathbf{w}^T\mathbf{u})(\mathbf{u}^T\mathbf{J}\mathbf{w}) \\
&= -(\mathbf{w}^T\mathbf{J}^T\mathbf{u})(\mathbf{u}^T\mathbf{w}) + (\mathbf{w}^T\mathbf{u})(\mathbf{u}^T\mathbf{J}\mathbf{w}) \\
&= -[(\mathbf{J}\mathbf{w})^T\mathbf{u}](\mathbf{w}^T\mathbf{u}) + (\mathbf{w}^T\mathbf{u})[\mathbf{u}^T(\mathbf{J}\mathbf{w})] \\
&= -(\mathbf{w}_1^T\mathbf{u})(\mathbf{w}^T\mathbf{u}) + (\mathbf{w}^T\mathbf{u})(\mathbf{u}^T\mathbf{w}_1) \\
&= 0. \qquad (4.42)
\end{aligned}$$

Now, after expanding along the third row of (4.40), we have

$$\begin{aligned}
\boldsymbol{n}_{\mathbf{w}}^T\mathbf{w}^+ + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta &= \boldsymbol{n}_{\mathbf{w}}^T(\mathbf{w}+\Delta\mathbf{w}) + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta \\
&= \boldsymbol{n}_{\mathbf{w}}^T\mathbf{w} + \underbrace{\left(\boldsymbol{n}_{\mathbf{w}}^T\Delta\mathbf{w} + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta\right)}_{=0} \\
&= \boldsymbol{n}_{\mathbf{w}}^T\mathbf{w} \\
&= 0.
\end{aligned}$$

If we substitute the expression (4.27) for $\boldsymbol{n}_{\mathbf{w}}$ and (4.38) for $\mathbf{w}^+$ into the left hand side, then one obtains

$$\begin{aligned}
0 &= \boldsymbol{n}_{\mathbf{w}}^T\mathbf{w}^+ + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta \\
&= \left[\boldsymbol{n}_\alpha\mathbf{u}^T - \boldsymbol{n}_\beta(\mathbf{J}\mathbf{u})^T\right]\left[\Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}\right] + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta. \qquad (4.43)
\end{aligned}$$

Furthermore, by expanding the first term on the right hand side, using the properties of $\mathbf{J}$, then

$$\begin{aligned}
\left[\boldsymbol{n}_\alpha\mathbf{u}^T - \boldsymbol{n}_\beta(\mathbf{J}\mathbf{u})^T\right](\Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}) &= \boldsymbol{n}_\alpha\Delta\alpha\mathbf{u}^T\mathbf{u} + \boldsymbol{n}_\beta\Delta\beta\mathbf{u}^T\mathbf{J}^T\mathbf{J}\mathbf{u} \\
&= \boldsymbol{n}_\alpha\Delta\alpha\|\mathbf{u}\|^2 + \boldsymbol{n}_\beta\Delta\beta\|\mathbf{u}\|^2 \\
&= \left(\boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta\right)\|\mathbf{u}\|^2.
\end{aligned}$$

Consequently, (4.43) becomes

$$\left(\boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta\right)\|\mathbf{u}\|^2 + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta = (1+\|\mathbf{u}\|^2)(\boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta) = 0.$$

Observe that because $\mathbf{u}$ is real, $(1 + \|\mathbf{u}\|^2)$ is nonzero. Accordingly, after dividing both sides by $(1 + \|\mathbf{u}\|^2)$, then

$$\boldsymbol{n}_\alpha \Delta\alpha + \boldsymbol{n}_\beta \Delta\beta = 0. \tag{4.44}$$

We combine the two equations (4.41) and (4.44) below

$$\begin{bmatrix} \boldsymbol{n}_\beta & -\boldsymbol{n}_\alpha \\ \boldsymbol{n}_\alpha & \boldsymbol{n}_\beta \end{bmatrix} \begin{bmatrix} \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(\mathbf{w}^T\mathbf{w} + 1) \\ 0 \end{bmatrix}, \tag{4.45}$$

and compute $\Delta\alpha, \Delta\beta$ simultaneously. The matrix on the left hand side is always nonsingular except at the root (in which case all entries are zero). Observe that

$$\begin{aligned} \mathbf{w}^T\mathbf{J}^T\Delta\mathbf{w} &= -\mathbf{w}^T\mathbf{J}\Delta\mathbf{w} \\ &= -\mathbf{w}^T\mathbf{J}(\mathbf{w}^+ - \mathbf{w}) \\ &= -\mathbf{w}^T\mathbf{J}\mathbf{w}^+ + \mathbf{w}^T\mathbf{J}\mathbf{w} \\ &= -\mathbf{w}^T\mathbf{J}\mathbf{w}^+, \end{aligned}$$

where we have used the fact that $\mathbf{w}^T\mathbf{J}\mathbf{w} = 0$ for all $\mathbf{w}$, so that (4.44) can now be applied to simplify $\mathbf{w}^T\mathbf{J}^T\Delta\mathbf{w}$ as

$$\begin{aligned} \mathbf{w}^T\mathbf{J}^T\Delta\mathbf{w} &= -\mathbf{w}^T\mathbf{J}\mathbf{w}^+ \\ &= -\mathbf{w}^T\mathbf{J}(\Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}) \\ &= -\mathbf{w}^T(\Delta\alpha\mathbf{J}\mathbf{u} + \Delta\beta\mathbf{u}) \\ &= -\big[\Delta\alpha(\mathbf{w}^T\mathbf{J}\mathbf{u}) + \Delta\beta(\mathbf{w}^T\mathbf{u})\big] \\ &= -\big[\boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta\big] \\ &= 0. \end{aligned} \tag{4.46}$$

Notice that we have used the property $\mathbf{J}^2 = -\mathbf{I}_{2n}$ to arrive at the third to the last step above and the definition (4.38) for $\mathbf{w}^+$. Therefore, we have proved the key result

$$\mathbf{w}^T\mathbf{J}^T\Delta\mathbf{w} = 0.$$

The above analysis leads to the following fundamental result.

Theorem 4.4.1. *Let* $\boldsymbol{\phi}^{(k)} = [(\mathbf{J}\mathbf{w})^T, 0, 0]$ *be an approximation to the exact nullvector*

$\boldsymbol{n}^{(k)}$ *of*

$$\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)}) = \begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^T & 0 & 0 \end{bmatrix}.$$

(a). The matrix

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^T & 0 & 0 \\ (\mathbf{J}\mathbf{w})^T & 0 & 0 \end{bmatrix}, \tag{4.47}$$

is nonsingular at an algebraically simple eigenvalue of $\mathbf{A}\mathbf{z} = \lambda\mathbf{z}$.

(b). The (unique) solution of

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^T & 0 & 0 \\ (\mathbf{J}\mathbf{w})^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{w} \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} -\mathbf{M}\mathbf{w} \\ \frac{1}{2}(\mathbf{w}^T\mathbf{w} - 1) \\ 0 \end{bmatrix}, \tag{4.48}$$

is identical to the least squares solution of the under-determined system

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{w} \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} -\mathbf{M}\mathbf{w} \\ \frac{1}{2}(\mathbf{w}^T\mathbf{w} - 1) \end{bmatrix}. \tag{4.49}$$

Proof:

(a). At the root $\boldsymbol{\phi} = \boldsymbol{n}$ and since the real $(2n+1)$ by $(2n+2)$ Jacobian (4.9) has been shown to be of full rank in Theorem 4.2.1, so adding the $(2n+2)$th row, $\boldsymbol{n}^T$ to the Jacobian (4.9) increases the row rank by one (since the nullvector, $\boldsymbol{n}$ is orthogonal to every row of $\mathbf{F}_{\mathbf{v}}(\mathbf{v})$). Hence,

$$\text{rank}\left(\begin{bmatrix} \mathbf{F}_{\mathbf{v}}(\mathbf{v}) \\ \boldsymbol{n}^T \end{bmatrix}\right) = 2n + 2.$$

Hence, the matrix in (4.47) is nonsingular at the root.

(b). Recall that $\Delta\mathbf{v}^{(k)} = [\Delta\mathbf{w}^T, \Delta\alpha, \Delta\beta]$. By using (4.46), this implies

$$\boldsymbol{\phi}^{(k)^T}\Delta\mathbf{v}^{(k)} = (\mathbf{J}\mathbf{w})^T\Delta\mathbf{w} = \mathbf{w}^T\mathbf{J}^T\Delta\mathbf{w} = 0. \tag{4.50}$$

This shows that (4.48) and (4.49) are equivalent.

■

The above result means that instead of solving (4.19) or (4.49) via QR factorisation at a cost of approximately $\frac{32}{3}n^3$ floating point operations, we could use LU factorisation to solve (4.48) more efficiently at a cost of approximately $\frac{16}{3}n^3$.

We now present Algorithm 15 for computing the algebraically simple complex eigenpair of $\mathbf{A}$.

Algorithm 15 Eigenpair Computation using Newton's method

Input: $\mathbf{A}, \mathbf{w}^{(0)} = [\mathbf{z}_1^{(0)}, \mathbf{z}_2^{(0)}], \mathbf{v}^{(0)} = [\mathbf{w}^{(0)}, \alpha^{(0)}, \beta^{(0)}]^T, k_{\max}$ and tol.

1: **for** $k = 0, 1, 2, \ldots$ until convergence **do**

2: Compute the LU factorisation of

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^T & 0 & 0 \\ (\mathbf{J}\mathbf{w})^T & 0 & 0 \end{bmatrix}.$$

3: Form

$$\mathbf{d}^{(k)} = \begin{bmatrix} -\mathbf{M}\mathbf{w} \\ \frac{1}{2}(\mathbf{w}^T\mathbf{w} - 1) \\ 0 \end{bmatrix}.$$

4: Solve the lower triangular system $L\mathbf{c}^{(k)} = \mathbf{d}^{(k)}$ for $\mathbf{c}^{(k)}$.

5: Solve the upper triangular system $U\Delta\mathbf{v}^{(k)} = \mathbf{c}^{(k)}$ for $\Delta\mathbf{v}^{(k)}$.

6: Update $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta\mathbf{v}^{(k)}$.

7: **end for**

Output: $\mathbf{v}^{(k_{\max})} = [\mathbf{w}^{(k_{\max})}, \alpha^{(k_{\max})}, \beta^{(k_{\max})}]^T$.

Stop Algorithm 15 as soon as

$$\|\Delta\mathbf{v}^{(k)}\| \leq \text{tol}.$$

Example 4.4.1. *We consider the same example as in Example 4.2.1, with the same starting guesses but with a different algorithm: Algorithm 15. We stopped Algorithm 15, when*

$$\|\Delta\mathbf{v}^{(k)}\| \leq 5.6 \times 10^{-14}.$$

The results of Table 4.2 agree with those of Table 4.1 but with little disparities in the last two columns. This indeed show that the solution obtained by solving the underdetermined system (4.19) *is equivalent to those obtained by solving the square system*

k	$\alpha^{(k)}$	$\beta^{(k)}$	$\|\mathbf{w}^{(k+1)} - \mathbf{w}^{(k)}\|$	$\|\lambda^{(k+1)} - \lambda^{(k)}\|$	$\|\Delta\mathbf{v}^{(k)}\|$	$\|\mathbf{F}(\mathbf{v}^{(k)})\|$
0	0.00000e+00	2.50000	3.8e+00	7.8e-01	3.9e+00	3.6e+01
1	2.34253e-01	1.75371	1.8e+00	2.2e-01	1.8e+00	7.8e+00
2	1.18745e-01	1.94460	8.1e-01	1.4e-01	8.2e-01	1.7e+00
3	4.47044e-02	2.06484	2.5e-01	7.0e-02	2.6e-01	3.4e-01
4	8.82702e-03	2.12479	3.1e-02	1.7e-02	3.5e-02	3.7e-02
5	2.48114e-04	2.13905	4.8e-04	5.2e-04	7.1e-04	7.1e-04
6	1.80714e-05	2.13950	1.2e-07	2.5e-07	2.8e-07	2.8e-07
7	1.81999e-05	2.13950	1.3e-14	8.4e-14	8.5e-14	6.3e-14
8	1.81999e-05	2.13950	1.0e-14	4.8e-14	4.9e-14	5.3e-14

Table 4.2: Values of $\alpha^{(k)}$ and $\beta^{(k)}$ of Example 4.4.1. Columns 5 and 6 show that the results converged quadratically for $k = 3, 4, 5, 6$ and 7.

(4.48), *the disparities in the eigth and nineth rows are caused by round off errors. It also shows that Algorithm 14 and Algorithm 15 are equivalent which is our aim.*

4.5 Computing the Eigenpairs $(\mathbf{z}, \lambda)$ by solving a Square Complex System of Equations for $\mathbf{B} = \mathbf{I}$

In this section, our emphasis will be to compute the eigenpairs $(\mathbf{z}, \lambda)$ from the eigenvalue problem $\mathbf{Az} = \lambda\mathbf{z}$ in complex arithmetic rather than real arithmetic. To this end, we will rewrite the $(2n + 2)$ real equations (4.48) back in complex form. This will yield $(n + 1)$ complex equations in $(n + 1)$ complex unknowns. It turns out that the system we just derived is precisely what we would have derived if we had ignored the non uniqueness and non differentiability questions about the normalisation

$$-\frac{1}{2}\mathbf{z}^H\mathbf{z} + \frac{1}{2} = 0,$$

and 'blindly' applied Newton's method to (4.4) with $\mathbf{B} = \mathbf{I}$. Since the cost of solving a complex linear system of equations is roughly three times what it takes to solve a real system, this means that this method will cost approximately n^3 floating point operations when a solver like LU factorisation is used to solve the complex system. This should be compared with the cost of solving the $(2n + 2)$ by $(2n + 2)$ square system (4.48) which has an approximate cost of $\frac{2}{3}(2n)^3 = \frac{16}{3}n^3$ floating point operations.

The plan of this section is as follows. We begin by deriving the $(n+1)$ complex equation in $(n+1)$ complex unknowns from (4.48). It should be remarked that since (4.48) is a nonsingular system of equations, writing it in complex form must also produce a nonsingular system.

Recall that by using (4.9) with $\mathbf{B} = \mathbf{I}$, the expanded form of $\mathbf{F_v}(\mathbf{v})\Delta\mathbf{v} = -\mathbf{F}(\mathbf{v})$ in (4.19) is

$$\begin{bmatrix} (\mathbf{A}-\alpha\mathbf{I}) & \beta\mathbf{I} & -\mathbf{z}_1 & \mathbf{z}_2 \\ -\beta\mathbf{I} & (\mathbf{A}-\alpha\mathbf{I}) & -\mathbf{z}_2 & -\mathbf{z}_1 \\ -\mathbf{z}_1^T & -\mathbf{z}_2^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{z}_1 \\ \Delta\mathbf{z}_2 \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = - \begin{bmatrix} (\mathbf{A}-\alpha\mathbf{I})\mathbf{z}_1 + \beta\mathbf{z}_2 \\ -\beta\mathbf{z}_1 + (\mathbf{A}-\alpha\mathbf{I})\mathbf{z}_2 \\ -\frac{1}{2}(\mathbf{z}_1^T\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{z}_2) + \frac{1}{2} \end{bmatrix}. \tag{4.51}$$

But, we know from (4.12) that $\mathbf{w} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix}$ and $\mathbf{Jw} = \begin{bmatrix} \mathbf{z}_2 \\ -\mathbf{z}_1 \end{bmatrix}$. So that

$$[(\mathbf{Jw})^T, 0, 0] = [\mathbf{z}_2^T, -\mathbf{z}_1^T, 0, 0].$$

Moreover,

$$\mathbf{w}^T\mathbf{w} = \mathbf{z}_1^T\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{z}_2,$$

and from (4.10),

$$\mathbf{M} = \begin{bmatrix} (\mathbf{A}-\alpha\mathbf{I}) & \beta\mathbf{I} \\ -\beta\mathbf{I} & (\mathbf{A}-\alpha\mathbf{I}) \end{bmatrix}.$$

The above relationships means that we can rewrite (4.48) as

$$\begin{bmatrix} (\mathbf{A}-\alpha\mathbf{I}) & \beta\mathbf{I} & -\mathbf{z}_1 & \mathbf{z}_2 \\ -\beta\mathbf{I} & (\mathbf{A}-\alpha\mathbf{I}) & -\mathbf{z}_2 & -\mathbf{z}_1 \\ -\mathbf{z}_1^T & -\mathbf{z}_2^T & 0 & 0 \\ \mathbf{z}_2^T & -\mathbf{z}_1^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{z}_1 \\ \Delta\mathbf{z}_2 \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = - \begin{bmatrix} (\mathbf{A}-\alpha\mathbf{I})\mathbf{z}_1 + \beta\mathbf{z}_2 \\ -\beta\mathbf{z}_1 + (\mathbf{A}-\alpha\mathbf{I})\mathbf{z}_2 \\ -\frac{1}{2}(\mathbf{z}_1^T\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{z}_2) + \frac{1}{2} \\ 0 \end{bmatrix}. \tag{4.52}$$

We state the following result.

Lemma 4.5.1. *The square $(2n+2)$ by $(2n+2)$ system of real equations* (4.48) *is equivalent to the $(n+1)$ by $(n+1)$ system of complex equations*

$$\begin{bmatrix} (\mathbf{A}-\lambda\mathbf{I}) & -\mathbf{z} \\ -\mathbf{z}^H & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{z} \\ \Delta\lambda \end{bmatrix} = - \begin{bmatrix} (\mathbf{A}-\lambda\mathbf{I})\mathbf{z} \\ -\frac{1}{2}\mathbf{z}^H\mathbf{z} + \frac{1}{2} \end{bmatrix}. \tag{4.53}$$

Proof: Note that the first two rows on the left hand side of (4.52) are the real and imaginary parts of

$$(\mathbf{A} - \lambda\mathbf{I})\Delta\mathbf{z} - \Delta\lambda\mathbf{z},$$

that is, if we expand the first row on the left hand side of (4.53), using $\mathbf{z} = \mathbf{z}_1 + i\mathbf{z}_2$, $\lambda = \alpha + i\beta$ and $\Delta\lambda = \Delta\alpha + i\Delta\beta$. The last two rows on the left hand side of (4.52) are the real and imaginary parts of $-\mathbf{z}^H\Delta\mathbf{z}$. The same argument holds for the right hand sides of (4.52) and (4.53). ∎

Therefore,

$$\begin{bmatrix} \mathbf{A} - \lambda\mathbf{I} & -\mathbf{z} \\ -\mathbf{z}^H & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{z} \\ \Delta\lambda \end{bmatrix} = - \begin{bmatrix} (\mathbf{A} - \lambda\mathbf{I})\mathbf{z} \\ -\frac{1}{2}\mathbf{z}^H\mathbf{z} + \frac{1}{2} \end{bmatrix},$$

and

$$\begin{bmatrix} (\mathbf{A} - \alpha\mathbf{I}) & \beta\mathbf{I} & -\mathbf{z}_1 & \mathbf{z}_2 \\ -\beta\mathbf{I} & (\mathbf{A} - \alpha\mathbf{I}) & -\mathbf{z}_2 & -\mathbf{z}_1 \\ -\mathbf{z}_1^T & -\mathbf{z}_2^T & 0 & 0 \\ \mathbf{z}_2^T & -\mathbf{z}_1^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{z}_1 \\ \Delta\mathbf{z}_2 \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = - \begin{bmatrix} (\mathbf{A} - \alpha\mathbf{I})\mathbf{z}_1 + \beta\mathbf{z}_2 \\ -\beta\mathbf{z}_1 + (\mathbf{A} - \alpha\mathbf{I})\mathbf{z}_2 \\ -\frac{1}{2}(\mathbf{z}_1^T\mathbf{z}_1 + \mathbf{z}_2^T\mathbf{z}_2) + \frac{1}{2} \\ 0 \end{bmatrix},$$

are equivalent. More importantly, the last system of equations is the same as (4.48) *i.e.*,

$$\begin{bmatrix} \mathbf{M} & -\mathbf{w} & \mathbf{J}\mathbf{w} \\ -\mathbf{w}^T & 0 & 0 \\ (\mathbf{J}\mathbf{w})^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{w} \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} -\mathbf{M}\mathbf{w} \\ \frac{1}{2}(\mathbf{w}^T\mathbf{w} - 1) \\ 0 \end{bmatrix}. \tag{4.54}$$

Next, we present Algorithm 16 for computing the complex eigenpair of $\mathbf{A}$ using complex arithmetic.

Stop Algorithm 16 as soon as

$$\|\Delta\mathbf{v}^{(k)}\| \leq tol.$$

Example 4.5.1. *We consider the same example as in Example 4.2.1, with the same starting guesses but with Algorithm 16. We stopped Algorithm 16, when*

$$\|\Delta\mathbf{v}^{(k)}\| \leq 5.6 \times 10^{-14}.$$

Algorithm 16 Eigenpair Computation using Newton's method

Input: $\mathbf{A}, \mathbf{v}^{(0)} = [\mathbf{z}_1^{(0)}, \mathbf{z}_2^{(0)}, \alpha^{(0)}, \beta^{(0)}]^T, k_{\max}$ and tol.

1: **for** $k = 0, 1, 2, \ldots,$ until convergence **do**

2: Compute the LU factorisation of

$$\begin{bmatrix} \mathbf{A} - \lambda^{(k)}\mathbf{I} & -\mathbf{z}^{(k)} \\ -(\mathbf{z}^{(k)})^H & 0 \end{bmatrix}.$$

3: Form

$$\mathbf{d}^{(k)} = -\begin{bmatrix} (\mathbf{A} - \lambda^{(k)}\mathbf{I})\mathbf{z}^{(k)} \\ -\frac{1}{2}\mathbf{z}^{(k)^H}\mathbf{z}^{(k)} + \frac{1}{2} \end{bmatrix}.$$

4: Solve the lower triangular system $L\mathbf{y}^{(k)} = \mathbf{d}^{(k)}$ for $\mathbf{y}^{(k)}$.

5: Solve the upper triangular system $U\Delta\mathbf{v}^{(k)} = \mathbf{y}^{(k)}$ for $\Delta\mathbf{v}^{(k)}$.

6: Update $\mathbf{v}^{(k+1)} = \mathbf{v}^{(k)} + \Delta\mathbf{v}^{(k)}$.

7: **end for**

Output: $\mathbf{v}^{(k_{\max})}$.

Computed eigenpairs are shown in Table 4.3. Observe that we obtained quadratic

k	$\alpha^{(k)} + i\beta^{(k)}$	$\|\mathbf{z}^{(k+1)} - \mathbf{z}^{(k)}\|$	$\|\lambda^{(k+1)} - \lambda^{(k)}\|$	$\|\Delta\mathbf{v}^{(k)}\|$	$\|\mathbf{F}(\mathbf{v}^{(k)})\|$
0	0.00000e+00+2.50000e+00i	3.8e+00	7.8e-01	3.9e+00	3.6e+01
1	2.34253e-01+1.75371e+00i	1.8e+00	2.2e-01	1.8e+00	7.8e+00
2	1.18745e-01+1.94460e+00i	8.1e-01	1.4e-01	8.2e-01	1.7e+00
3	4.47044e-02+2.06484e+00i	2.5e-01	7.0e-02	2.6e-01	3.4e-01
4	8.82702e-03+2.12479e+00i	3.1e-02	1.7e-02	3.5e-02	3.7e-02
5	2.48114e-04+2.13905e+00i	4.8e-04	5.2e-04	7.1e-04	7.1e-04
6	1.80714e-05+2.13950e+00i	1.2e-07	2.5e-07	2.8e-07	2.8e-07
7	1.81999e-05+2.13950e+00i	1.1e-14	3.7e-14	3.8e-14	6.3e-14

Table 4.3: Values of $\alpha^{(k)}$ and $\beta^{(k)}$ of Example 4.5.1. Columns 5 and 6 show that the results converged quadratically for $k = 3, 4, 5, 6$ and 7.

convergence from the last two columns of Table 4.3 for $k = 3, 4, 5, 6$ and 7. Figure 4-1 shows a plot of the logarithm of the residuals against the number of iterations and the quadratic convergence of the residuals. One or two steps of iterative refinements did not improve the results either. At the root, the condition number of $\mathbf{F}_{\mathbf{v}}(\mathbf{v}^{(k)})$ is approximately 3×10^3. As predicted by the theory, the results of Table 4.3 tallies with those of Tables 4.1 and 4.2.

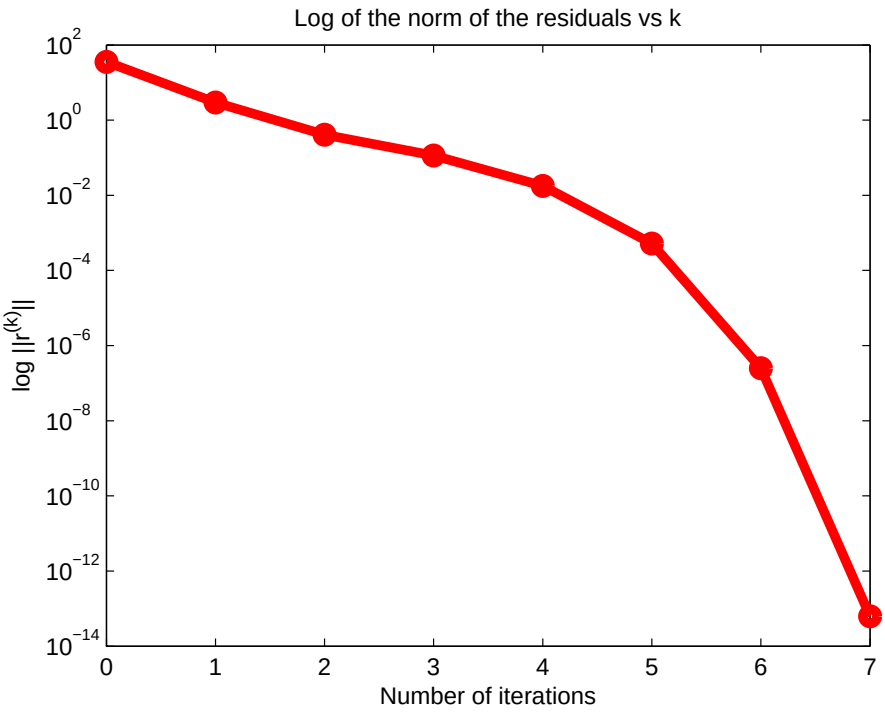

Figure 4-1: Convergence history of the eigenvalue residuals on Example 4.5.1. The figure shows that the residual converge quadratically.

4.6 Square System of Equations for The Numerical Computation of the Complex Eigenpairs of the Pencil $(\mathbf{A}, \mathbf{B})$ for $\mathbf{B} \neq \mathbf{I}$

The focus in this section is to present the theory discussed in Section 4.4 for the general case in which $\mathbf{B} \neq \mathbf{I}$. The main conclusion is that unlike the $\mathbf{B} = \mathbf{I}$ case where we had the nice result $\mathbf{w}^T\mathbf{J}^T\Delta\mathbf{w} = 0$, there is does not appear to be an equivalent orthogonality result for the $\mathbf{B} \neq \mathbf{I}$ case.

First of all, we revise important formulae from Section 4.3 that will be of use in this section, which are (4.27) and (4.28)

$$\boldsymbol{n}_{\mathbf{w}} = \boldsymbol{n}_{\alpha}\mathbf{u} - \boldsymbol{n}_{\beta}\mathbf{J}\mathbf{u},$$

and

$$\boldsymbol{n}_\alpha = \mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u}, \quad \text{with} \quad \boldsymbol{n}_\beta = \mathbf{w}^T\mathbf{B}_2\mathbf{u},$$

so that

$$\boldsymbol{n} = [\boldsymbol{n}_\mathbf{w}^T, \boldsymbol{n}_\alpha, \boldsymbol{n}_\beta] = [(\boldsymbol{n}_\alpha\mathbf{u} - \boldsymbol{n}_\beta\mathbf{J}\mathbf{u})^T, (\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u}), (\mathbf{w}^T\mathbf{B}_2\mathbf{u})].$$

Consider the problem of solving the under-determined linear system of equations (4.31) for the $2n+2$ real unknowns $\Delta\mathbf{v} = [\Delta\mathbf{w}^T, \Delta\alpha, \Delta\beta]$. It was stated in Lemma 1.5.1 that the minimum norm solution to an under-determined linear system of equations is orthogonal to the nullspace. It is an application of this result that yields the following important relationship:

$$0 = \boldsymbol{n}^T\Delta\mathbf{v} = \boldsymbol{n}_\mathbf{w}^T\Delta\mathbf{w} + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta. \tag{4.55}$$

If we add the nullvector $\boldsymbol{n}$ to the last row of (4.31) with $\mathbf{B} \neq \mathbf{I}$ and using (4.55), then

$$\begin{bmatrix} \mathbf{M} & -\mathbf{B}_2\mathbf{w} & \mathbf{B}_2\mathbf{J}\mathbf{w} \\ -(\mathbf{B}_2\mathbf{w})^T & 0 & 0 \\ \boldsymbol{n}_\mathbf{w}^T & \boldsymbol{n}_\alpha & \boldsymbol{n}_\beta \end{bmatrix} \begin{bmatrix} \Delta\mathbf{w} \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} -\mathbf{M}\mathbf{w} \\ \frac{1}{2}(\mathbf{w}^T\mathbf{B}_2\mathbf{w} - 1) \\ 0 \end{bmatrix}. \tag{4.56}$$

We now rewrite (4.19) in expanded form as:

$$\begin{bmatrix} \mathbf{M} & -\mathbf{B}_2\mathbf{w} & \mathbf{B}_2\mathbf{J}\mathbf{w} \\ -(\mathbf{B}_2\mathbf{w})^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{w} \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = -\begin{bmatrix} \mathbf{M}\mathbf{w} \\ -\frac{1}{2}\mathbf{w}^T\mathbf{B}_2\mathbf{w} + \frac{1}{2} \end{bmatrix}, \tag{4.57}$$

or,

$$\begin{aligned} \mathbf{M}\Delta\mathbf{w} - \Delta\alpha\mathbf{B}_2\mathbf{w} + \Delta\beta\mathbf{B}_2\mathbf{J}\mathbf{w} &= -\mathbf{M}\mathbf{w} \\ -\mathbf{w}^T\mathbf{B}_2\Delta\mathbf{w} &= \frac{1}{2}\mathbf{w}^T\mathbf{B}_2\mathbf{w} - \frac{1}{2}. \end{aligned}$$

After rearrangement, the first equation reduces to

$$\mathbf{M}\mathbf{w}^+ - \Delta\alpha\mathbf{B}_2\mathbf{w} + \Delta\beta\mathbf{B}_2\mathbf{J}\mathbf{w} = \mathbf{0}. \tag{4.58}$$

By multiplying both sides of the second equation by 2, we obtain:

$$2\mathbf{w}^T\mathbf{B}_2\Delta\mathbf{w} + \mathbf{w}^T\mathbf{B}_2\mathbf{w} = 1.$$

This in turn reduces to

$$\mathbf{w}^T\mathbf{B}_2(\mathbf{w} + 2\Delta\mathbf{w}) = 1. \tag{4.59}$$

Since $\mathbf{w}^+ = \mathbf{w} + \Delta\mathbf{w}$, $2\Delta\mathbf{w} = 2\mathbf{w}^+ - 2\mathbf{w}$ and $\mathbf{w} + 2\Delta\mathbf{w} = 2\mathbf{w}^+ - \mathbf{w}$, then $\mathbf{w}^T\mathbf{B}_2(\mathbf{w} + 2\Delta\mathbf{w}) = \mathbf{w}^T\mathbf{B}_2(2\mathbf{w}^+ - \mathbf{w}) = 2\mathbf{w}^T\mathbf{B}_2\mathbf{w}^+ - \mathbf{w}^T\mathbf{B}_2\mathbf{w}$. Consequently,

$$\mathbf{w}^T\mathbf{B}_2\mathbf{w}^+ = \frac{1}{2}(\mathbf{w}^T\mathbf{B}_2\mathbf{w} + 1). \tag{4.60}$$

The combined set of equations (4.58) and (4.60), which is the simplified form of (4.57), can be expressed as:

$$\begin{bmatrix} \mathbf{M} & -\mathbf{B}_2\mathbf{w} & \mathbf{B}_2\mathbf{J}\mathbf{w} \\ -(\mathbf{B}_2\mathbf{w})^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w}^+ \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ -\frac{1}{2}(\mathbf{w}^T\mathbf{B}_2\mathbf{w} + 1) \end{bmatrix}. \tag{4.61}$$

Now, if we expand along the first row of (4.61), then

$$\mathbf{M}\mathbf{w}^+ = \Delta\alpha\mathbf{B}_2\mathbf{w} - \Delta\beta\mathbf{B}_2\mathbf{J}\mathbf{w}. \tag{4.62}$$

This means that we could solve (4.61) by solving

$$\mathbf{M}\mathbf{u} = \mathbf{B}_2\mathbf{w}, \text{ and } \mathbf{M}\mathbf{J}\mathbf{u} = \mathbf{J}\mathbf{B}_2\mathbf{w}, \text{ (by property 6 of } \mathbf{J}) \tag{4.63}$$

for $\mathbf{u}$, after which the solution of (4.62) is given by

$$\mathbf{w}^+ = \Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}. \tag{4.64}$$

By adding $\boldsymbol{n}$ to the last row of (4.61), we have

$$\begin{bmatrix} \mathbf{M} & -\mathbf{B}_2\mathbf{w} & \mathbf{B}_2\mathbf{J}\mathbf{w} \\ (\mathbf{B}_2\mathbf{w})^T & 0 & 0 \\ \boldsymbol{n}_\mathbf{w}^T & \boldsymbol{n}_\alpha & \boldsymbol{n}_\beta \end{bmatrix} \begin{bmatrix} \mathbf{w}^+ \\ \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \frac{1}{2}(\mathbf{w}^T\mathbf{B}_2\mathbf{w} + 1) \\ \boldsymbol{n}_\mathbf{w}^T\mathbf{w} \end{bmatrix}. \tag{4.65}$$

Now, expand along the middle row of (4.65),

$$\mathbf{w}^T\mathbf{B}_2\mathbf{w}^+ = \frac{1}{2}(\mathbf{w}^T\mathbf{B}_2\mathbf{w} + 1),$$

and from (4.64), $\mathbf{w}^+ = \Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}$, where $\mathbf{u}$ is given by (4.63). This implies that by taking the inner product of both sides with $\mathbf{w}$, yields

$$\mathbf{w}^T\mathbf{B}_2\mathbf{w}^+ = \Delta\alpha(\mathbf{w}^T\mathbf{B}_2\mathbf{u}) - \Delta\beta(\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u}) = \frac{1}{2}(\mathbf{w}^T\mathbf{B}_2\mathbf{w} + 1).$$

Using the definition (4.28) for $\boldsymbol{n}_\alpha$ and $\boldsymbol{n}_\beta$ with $\mathbf{B} \neq \mathbf{I}$, we obtain

$$\boldsymbol{n}_\beta\Delta\alpha - \boldsymbol{n}_\alpha\Delta\beta = \frac{1}{2}(\mathbf{w}^T\mathbf{B}_2\mathbf{w} + 1), \tag{4.66}$$

where the unknown quantities $\Delta\alpha$ and $\Delta\beta$ are to be determined, so we need an extra equation to be able to do so. Note that by using $\boldsymbol{n}_\mathbf{w} = \boldsymbol{n}_\alpha\mathbf{u} - \boldsymbol{n}_\beta\mathbf{J}\mathbf{u}$, and (4.28) we can simplify

$$\begin{aligned}
\boldsymbol{n}_\mathbf{w}^T\mathbf{w} &= \boldsymbol{n}_\alpha\mathbf{u}^T\mathbf{w} - \boldsymbol{n}_\beta\mathbf{u}^T\mathbf{J}^T\mathbf{w} \\
&= \boldsymbol{n}_\alpha\mathbf{u}^T\mathbf{w} + \boldsymbol{n}_\beta\mathbf{u}^T\mathbf{J}\mathbf{w} \\
&= (\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u})(\mathbf{u}^T\mathbf{w}) + (\mathbf{w}^T\mathbf{B}_2\mathbf{u})(\mathbf{u}^T\mathbf{J}\mathbf{w}).
\end{aligned} \tag{4.67}$$

Now, after expanding along the third row of (4.65), we obtain

$$\begin{aligned}
\boldsymbol{n}_\mathbf{w}^T\mathbf{w}^+ + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta &= \boldsymbol{n}_\mathbf{w}^T(\mathbf{w} + \Delta\mathbf{w}) + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta \\
&= \boldsymbol{n}_\mathbf{w}^T\mathbf{w} + \underbrace{\left(\boldsymbol{n}_\mathbf{w}^T\Delta\mathbf{w} + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta\right)}_{=0} \\
&= \boldsymbol{n}_\mathbf{w}^T\mathbf{w}.
\end{aligned}$$

If we substitute the expression (4.27) for $\boldsymbol{n}_\mathbf{w}$ and (4.64) for $\mathbf{w}^+$ into the left hand side, then one obtains

$$\begin{aligned}
\boldsymbol{n}_\mathbf{w}^T\mathbf{w}^+ + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta &= \left[\boldsymbol{n}_\alpha\mathbf{u}^T - \boldsymbol{n}_\beta(\mathbf{J}\mathbf{u})^T\right]\left[\Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}\right] + \boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta \\
&= \boldsymbol{n}_\mathbf{w}^T\mathbf{w}.
\end{aligned} \tag{4.68}$$

Furthermore, by expanding the first term on the right hand side, using the

properties of $\mathbf{J}$, then

$$\begin{aligned}\left[\boldsymbol{n}_\alpha \mathbf{u}^T - \boldsymbol{n}_\beta (\mathbf{J}\mathbf{u})^T\right](\Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}) &= \boldsymbol{n}_\alpha \Delta\alpha \mathbf{u}^T\mathbf{u} + \boldsymbol{n}_\beta \Delta\beta \mathbf{u}^T\mathbf{J}^T\mathbf{J}\mathbf{u} \\ &= \boldsymbol{n}_\alpha \Delta\alpha \|\mathbf{u}\|^2 + \boldsymbol{n}_\beta \Delta\beta \|\mathbf{u}\|^2.\end{aligned}$$

Consequently, (4.68) becomes

$$(\boldsymbol{n}_\alpha \Delta\alpha + \boldsymbol{n}_\beta \Delta\beta)\|\mathbf{u}\|^2 + (\boldsymbol{n}_\alpha \Delta\alpha + \boldsymbol{n}_\beta \Delta\beta) = (1 + \|\mathbf{u}\|^2)(\boldsymbol{n}_\alpha \Delta\alpha + \boldsymbol{n}_\beta \Delta\beta) = \boldsymbol{n}_{\mathbf{w}}^T \mathbf{w}.$$

Observe that because $\mathbf{u}$ is real, $(1 + \|\mathbf{u}\|^2)$ is nonzero. Accordingly, after dividing both sides by $(1 + \|\mathbf{u}\|^2)$

$$\boldsymbol{n}_\alpha \Delta\alpha + \boldsymbol{n}_\beta \Delta\beta = \frac{\boldsymbol{n}_{\mathbf{w}}^T \mathbf{w}}{(1 + \|\mathbf{u}\|^2)} = \frac{(\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u})(\mathbf{u}^T\mathbf{w}) + (\mathbf{w}^T\mathbf{B}_2\mathbf{u})(\mathbf{u}^T\mathbf{J}\mathbf{w})}{(1 + \|\mathbf{u}\|^2)}. \quad (4.69)$$

We combine the two equations (4.66) and (4.69) below

$$\begin{bmatrix} \boldsymbol{n}_\beta & -\boldsymbol{n}_\alpha \\ \boldsymbol{n}_\alpha & \boldsymbol{n}_\beta \end{bmatrix} \begin{bmatrix} \Delta\alpha \\ \Delta\beta \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(\mathbf{w}^T\mathbf{B}_2\mathbf{w} + 1) \\ \frac{\boldsymbol{n}_{\mathbf{w}}^T \mathbf{w}}{(1+\|\mathbf{u}\|^2)} \end{bmatrix}, \quad (4.70)$$

and compute $\Delta\alpha$, $\Delta\beta$ simultaneously. The matrix on the left hand side is always nonsingular except at the root, this is because its determinant is $\boldsymbol{n}_\alpha^2 + \boldsymbol{n}_\beta^2$. Equation (4.69) can now be applied to simplify

$$\begin{aligned}\mathbf{w}^T\mathbf{J}^T\mathbf{B}_2\Delta\mathbf{w} &= -\mathbf{w}^T\mathbf{J}\mathbf{B}_2\Delta\mathbf{w} \\ &= -\mathbf{w}^T\mathbf{B}_2\mathbf{J}\Delta\mathbf{w} \\ &= -\mathbf{w}^T\mathbf{B}_2\mathbf{J}(\mathbf{w}^+ - \mathbf{w}) \\ &= -\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{w}^+ + \mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{w} \\ &= -\mathbf{w}^T\mathbf{B}_2\mathbf{J}(\Delta\alpha\mathbf{u} - \Delta\beta\mathbf{J}\mathbf{u}) \\ &= -\mathbf{w}^T\mathbf{B}_2(\Delta\alpha\mathbf{J}\mathbf{u} + \Delta\beta\mathbf{u}) \\ &= -\left[\Delta\alpha(\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u}) + \Delta\beta(\mathbf{w}^T\mathbf{B}_2\mathbf{u})\right] \\ &= -\left[\boldsymbol{n}_\alpha\Delta\alpha + \boldsymbol{n}_\beta\Delta\beta\right] \\ &= -\frac{\boldsymbol{n}_{\mathbf{w}}^T\mathbf{w}}{(1 + \|\mathbf{u}\|^2)} \\ &= -\left[(\mathbf{w}^T\mathbf{B}_2\mathbf{J}\mathbf{u})(\mathbf{u}^T\mathbf{w}) + (\mathbf{w}^T\mathbf{B}_2\mathbf{u})(\mathbf{u}^T\mathbf{J}\mathbf{w})\right]/(1 + \|\mathbf{u}\|^2). \quad (4.71)\end{aligned}$$

Notice that we have used the property $\mathbf{J}^2 = -\mathbf{I}_{2n}$ to arrive at the third to the last step above and the definition (4.38) for $\mathbf{w}^+$. In addition, by the property of $\Delta\mathbf{w}$, it tends to zero in the limit. This implies that $\mathbf{w}^T\mathbf{J}^T\mathbf{B}_2\Delta\mathbf{w} = 0$ in the limit. However, there is no reason to suppose that $\mathbf{w}^T\mathbf{J}^T\mathbf{B}_2\Delta\mathbf{w} = 0$, as the iteration converges. So there does not appear to be an analogue of the nice orthogonality result in Section 4.4 for $\mathbf{B} \neq \mathbf{I}$.

4.7 Conclusion

For the standard eigenvalue problem $\mathbf{A}\mathbf{z} = \lambda\mathbf{z}$, Ruhe [51, Section 3] used the normalisation $\mathbf{c}^H\mathbf{z} = 1$ and solved the resulting real $(n+1)$ by $(n+1)$ system of nonlinear equations to obtain $[\mathbf{z}, \lambda]^T$, we have been able to rigorously justify that, with the addition of the non differentiable normalisation $\mathbf{z}^H\mathbf{z} = 1$, it is still possible to convert the resulting system of under-determined linear equations into a nonsingular complex square one.

In this chapter, we have proved the mathematical equivalence of three algorithms viz-a-viz Algorithm 14, Algorithm 15 and Algorithm 16. The mathematical equivalence of the three algorithms means that the solution obtained by solving the under-determined linear system of equations is the same as those obtained by solving the square ones in the absence of round off errors. Numerical experiments are given which confirm the equivalence of the three algorithms.

CHAPTER 5

Conclusions and Further Work

In this thesis, we have studied the numerical solution of some linear and nonlinear eigenvalue problems. In particular, we have used Newton's method or its variants as well as extended versions of the implicit determinant method of Spence and Poulton [55] to achieve the following:

1. we have obtained an algorithm for computing when two eigenvalues coalesce in a parameter-dependent nonsymmetric matrix as the parameter is varied to form a 2-dimensional Jordan block in Chapter 2.

2. We have derived an efficient algorithm for computing a nearby defective matrix from a simple one which is cheaper and faster than earlier known ones.

3. We have contributed to a greater understanding of the natural normalisation for a complex eigenvector.

Future work might involve:

1. The extension of the techniques in Chapter 2 to compute more complicated Jordan structures, *e.g.*, 3-dimensional Jordan blocks, or Jordan blocks corresponding to eigenvalues of geometric multiplicities greater than one. The latter would require analogues of the "ABCD" Lemma using borderings of dimension greater than one.

2. One could extend the class of problems considered in Chapter 2, to more challenging physical problems, for example, the full linearized Navier-Stokes equations.

3. The use of more sophisticated nonlinear solvers than standard Newton's method in the solution of the nonlinear systems that arise in the nearby defective matrix problem. For example, one could consider the use of global Newton or optimization based algorithms to solve the nonlinear systems to compute the nearby defective matrix in Chapter 3.

BIBLIOGRAPHY

[1] *Block Thomas Algorithm*. *www4.nscu.edu/eos/users/w/white/www/white/ma580/chap2.5.pdf*.

[2] R. O. AKINOLA, M. A. FREITAG AND A. SPENCE, *The Calculation of the Distance to a Nearby Defective Matrix*, SIAM Journal of Matrix Analysis and Applications. (Submitted), (2009).

[3] R. ALAM, *On the Construction of Nearest Defective Matrices to a Normal Matrix*, Linear Algebra and its Applications, 395 (2005), pp. 367–370.

[4] R. ALAM, AND S. BORA, *On Sensitivity of Eigenvalues and Eigendecompositions of Matrices*, Linear Algebra and its Applications, 396 (2005), pp. 273–301.

[5] R. ALAM, S. BORA, R. BYERS, AND M. L. OVERTON, *Characterisation and Construction of the Nearest Defective Matrix via Coalescence of Pseudospectral Components*, Submitted: Linear Algebra and its Applications, (2009).

[6] K. E. ATKINSON, *An Introduction to Numerical Analysis*, John Wiley & Sons, Inc., 2nd ed., 1989.

[7] M. BENNANI, T. BRACONNIER, AND J. C. DUNYACH, *Solving Large-Scale Nonnormal Eigenproblems in the Aeronautical Industry using Parallel BLAS*, vol. 1, Proceedings of the International Conference and Exhi-

bition on High-Performance Computing and Networking Applications, April 1994, pp. 72–77.

[8] R. L. BISPLINGHOFF, AND H. ASHLEY, *Principles of Aeroelasticity*, John Wiley & Sons, New York, 1962, ch. 8.

[9] B. BOISVERT, R. POZO, K. REMINGTON, B. MILLER, AND R. LIPMAN, *Matrix Market*. `http://math.nist.gov/MatrixMarket/`.

[10] S. BOYD, *Lecture 8: Least Norm Solutions of Under-determined Equations*, EE263 Autumn 2008-09.

[11] N. CHEN, *Inverse Iteration on Defective Matrices*, Mathematics of Computation, 31 (1977), pp. 726–732.

[12] K. A. CLIFFE, A. SPENCE AND S. J. TAVENER, *The Numerical Analysis of Bifurcation Problems with Application to Fluid Mechanics*, in Acta Numerica, vol. 9, Cambridge University Press, 2000, pp. 40–131.

[13] J. W. DEMMEL, *A Numerical Analyst's Jordan Canonical Forms*, PhD thesis, University of California, Berkeley, 1983.

[14] ——, *Computing Stable Eigendecompositions of Matrices*, Linear Algebra and its Applications, 79 (1986), pp. 163–193.

[15] ——, *Applied Numerical Linear Algebra*, SIAM, 1997, ch. 3.

[16] J. E. DENNIS, JR, AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, no. 16 in Classics In Applied Mathematics, SIAM Philadelphia, 1996.

[17] P. DEUFLHARD, *Newton Methods for Nonlinear Problems*, Springer, 2004, ch. 4, pp. 174–175.

[18] P. DEUFLHARD, AND G. HEINDL, *Affine Invariant Convergence Theorems for Newton's Method and Extensions to Related Methods*, SIAM J. Numer. Anal., 16 (1979), pp. 1–10.

[19] I. DODSON, I. ZHANG, S. GREENE, H. ENGDAHL AND P. W. SAUER, *Is Modal Resonance a Precursur to Power System Oscillations?*, IEEE Transactions on Circuits and Systems, Part 1, 48 (2001), pp. 340–349.

[20] G. ENGELN-MULLGES, AND F. UHLIG, *Numerical Algorithms with C*, Springer, 1996.

[21] M. A. FREITAG, AND A. SPENCE, *Convergence of Inexact Inverse Iteration with Application to Preconditioned Iterative Solves*, BIT Numerical Mathematics, 47 (2006), pp. 27–44.

[22] ——, *The Calculation of the Distance to Instability by the Computation of a Jordan Block*, Submitted: Linear Algebra and its Application, (2009).

[23] G. H. GOLUB, AND C. F. VAN LOAN, *Matrix Computations*, The John Hopkins University Press, London, 3rd ed., 1996.

[24] G. H. GOLUB, AND Q. YE, *Inexact Inverse Iteration for Generalized Eigenvalue Problems*, BIT, 40 (2000), pp. 671–684.

[25] W. GOVAERTS, *Stable Solvers and Block Elimination for Bordered Ssytems*, SIAM J. Matrix Anal. Appl., 12 (1991), pp. 469–483.

[26] W. GOVAERTS, AND J. D. PRYCE, *Block Elimination with one Refinement Solves Bordered Linear Systems Accurately*, BIT, 30 (1990), pp. 490–507.

[27] ——, *Mixed Block Elimination for Linear Systems with Wider Borders*, IMA Journal of Numerical Analysis, 13 (1993), pp. 161–180.

[28] I. G. GRAHAM, A. SPENCE AND E. VAINIKKO, *Parallel Iterative Methods for Navier-Stokes Equations and Application to Eigenvalue Computation*, Concurrency and Computation: Practice and Experience, 15 (2003), pp. 1151–1168.

[29] K. K. GUPTA, *On a Numerical Solution of the Supersonic Panel Flutter Eigenproblem*, International Journal for Numerical Methods in Engineering, (1976), pp. 637–645.

[30] N. J. HIGHAM, *Accuracy and Stability of Numerical Algorithms*, SIAM, 1996, ch. 7.

[31] E. ISAACSON, AND H. B. KELLER, *Analysis of Numerical Methods*, John Wiley & Sons, Inc., London, 1966.

[32] A. JEFFREY, *Mathematics for Engineers and Scientists*, Nelson, 1969.

[33] H. B. KELLER, *Numerical Solution of Bifurcation and Nonlinear Eigenvalue Problems*, in Applications of Bifurcation Theory, in : P. Rabinowitz, ed., Academic Press, New York, 1977, pp. 359–384.

[34] A. N. KOLMOGOROV, AND S. V. FOMIN, *Introductory Real Analysis*, Dover Publications, Inc., 1970, ch. 2.

[35] E. KREYSZIG, *Advanced Engineering Mathematics*, John Wiley & Sons, Inc., New York, eighth ed., 1999.

[36] R. A. LIPPERT, AND A. EDELMAN, *The Computation and Sensitivity of Double Eigenvalues, in Advances in Computational Mathematics (Guangzhou, 1997)*, 202 in Lecture Notes in Pure and Applied Mathematics, Dekker, New York (1999), pp. 353–393.

[37] A. N. MALYSHEV, *A Formula for the 2-norm Distance from a Matrix to the Set of Matrices with Multiple Eigenvalues*, Numer. Math., 83 (1999), pp. 443–454.

[38] K. MEERBERGEN, AND D. ROOSE, *Matrix Transformations for Computing Rightmost Eigenvalues of Large Sparse Non-Symmetric Eigenvalue Problems*, IMA Journal of Numerical Analysis, 16 (1996), pp. 297–346.

[39] E. D. NERING, *Linear Algebra and Matrix Theory*, John Wiley & Sons, Inc., second ed., 1970, ch. 1, pp. 31–32.

[40] B. NOBLE, *Applied Linear Algebra*, Prentice-Hall, Inc., 1969, ch. 5, pp. 143–145.

[41] B. NOBLE, AND J. W. DANIEL, *Applied Linear Algebra*, Prentice-Hall, third ed., 1988, ch. 9, pp. 355–397.

[42] J. NOCEDAL, AND S. J. WRIGHT, *Numerical Optimization*, Springer, New York, 2nd ed., 2006.

[43] M. D. OLSON, *Finite Elements Applied to Panel Flutter*, AIAA, 5 (1967), pp. 2267–2270.

[44] ——, *Some Flutter Solutions using Finite Elements*, AIAA, 8 (1970), pp. 747–752.

[45] P. J. OLVER, AND C. SHAKIBAN, *Applied Linear Algebra*, Pearson Prentice Hall, 2006.

[46] J. M. ORTEGA, AND W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, London, 1970, ch. 8.

[47] M. L. OVERTON, *The Search for the Nearest Defective Matrix*, tech. rep., Courant Institute of Mathematical Sciences, New York University, 2006.

[48] B. N. PARLETT, AND Y. SAAD, *Complex Shift and Invert Strategies for Real Matrices*, Lin. Alg. Appl., (1987), pp. 575–595.

[49] W. C. RHEINBOLDT, *A Unified Convergence Theory for a Class of Iterative Processes*, SIAM J. Numer. Anal., 5 (1968).

[50] A. RUHE, *Properties of a Matrix with a Very Ill-Conditioned Eigenproblem*, Numerische Mathematik, 15 (1970), pp. 57–60.

[51] ——, *Algorithms for the Nonlinear Eigenvalue Problem*, SIAM J. Matrix Anal. Appl., 10 (1973), pp. 674–689.

[52] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, SIAM, 2nd ed., 2003, ch. 9.

[53] G. SANDER, C. BON AND M. GERADIN, *Finite Element Analysis of Supersonic Panel Flutter*, International Journal for Numerical Methods in Engineering, 7 (1973), pp. 379–394.

[54] A. SPENCE, AND C. POULTON, *Inverse Iteration for Nonlinear Eigenvalue Problems*, Electromagnetic Scattering-IUTAM Symposium on Asymptotics, Singularities and Homogenisation in Problems of Mechanics, 2003, pp. 585–594.

[55] ——, *Photonic Band Structure Calculations using Nonlinear Eigenvalue Techniques*, Journal of Computational Physics, 204 (2005), pp. 65 – 81.

[56] A. SPENCE, AND I. G. GRAHAM, *The Graduate Student's Guide to Numerical Analysis '98*, Springer, 1998, Lecture Notes from the VIII EPSRC Summer School in Numerical Analysis 3, pp. 176–216.

[57] G. W. STEWART, *Matrix Algorithms*, vol. II: Eigensystems, SIAM, 2001.

[58] J. G. SUN, *A Note on Simple Nonzero Singular Values*, J. Comput. Math., 6 (1988), pp. 258–266.

[59] F. TISSEUR, *Newton's Method in Floating Point Arithmetic and Iterative Refinement of Generalized Eigenvalue Problems*, SIAM J. Matrix Anal. Appl., 22 (2001), pp. 1038–1057.

[60] L. N. TREFETHEN, AND D. BAU III, *Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

[61] L. N. TREFETHEN, AND M. EMBREE, *Spectra and Pseudospectra*, Princeton University Press, 2005.

[62] J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Oxford University Press, Ely House, London W., 1965.

[63] ——, *Inverse Iteration in Theory and in Practice*, Istituto Nazionale di Alta Mathematica, Symposia Mathematica, *X* (1972), pp. 361–379.

[64] ——, *Note on Matrices with a very Ill-Conditioned Eigenproblem*, Numerische Mathematik, 19 (1972), pp. 176–178.

[65] ——, *Sensitivity of Eigenvalues ii*, Utilas Math., 25 (1984), pp. 5–76.

[66] ——, *Sensitivity of Eigenvalues ii*, Utilas Math., (1986), pp. 243–286.

[67] T. WRIGHT, *Eigtool*, tech. rep., Software available at, `http://www.comlab.ox.ac.uk/pseudospectra/eigtool`, 2002.